Reasoning about Structured Stochastic Systems in Continuous-Time

Thesis submitted for the degree of “Doctor of Philosophy”
by
Tal El-Hay

Submitted to the Senate of the Hebrew University
January 2011

This work was carried out under the supervision of

Prof. Nir Friedman

Abstract

Many real-life-tasks deal with dynamic systems that evolve in continuous-time. In many cases such systems are composed of many interacting components which evolve at various time scales. Thus, recent research on such multi-component systems is aimed at reasoning their dynamics directly in continuous-time.

A common approach to studying such systems is to construct a probabilistic model based on empirical observations. This approach requires three fundamental ingredients: a mathematical *modeling language* that captures the essential characteristics of a process in a compact manner; a *learning procedure*, allowing to estimate the structure and parameterization of a specific model from observations; and an *inference procedure* allowing to perform predictions given a model and observations as well as to compute the likelihood of the observations. A recent example are *Continuous-time Bayesian networks* (CTBNs) which are a compact representation of multi-component Markov processes with a sparse pattern of interactions. As research on such models is still in its early stages, there is a large gap between the wealth of modeling languages, learning procedures and inference algorithms available for more traditional domains and those available for continuous-time domain.

In this dissertation we enhance the range of applications amenable for continuous-time analysis by addressing these three issues. We begin by introducing a novel modeling language, *continuous-time Markov networks*, which allows learning a compact representation of the stationary distribution of a process. This modeling language is particularly suitable for learning biological sequence evolution, whose dynamics are dictated by an interplay between random mutations of individual components (nucleotides in DNA and RNA or amino-acids in proteins) and the global fitness of the resulting sequence. We derive an iterative procedure for learning this model from data, where each iteration requires solving the inference problem, which we show how to cast as a CTBN inference problem.

As the inference problem in CTBNs is crucial for learning both CTBNs and CTMNs from data, it is the focus of the rest of this dissertation, where we derive three different approximate algorithms that are complementary to each other. These algorithms adopt insights from existing state of the art methods for inference in finite dimensional domains while exploiting the continuous time representation to obtain efficient and relatively simple computations that naturally adapt to the dynamics of the process.

Our first inference algorithm is based on a Gibbs sampling strategy. This algorithm samples trajectories from the posterior distribution given the evidence and uses these samples to answer queries. We show how to perform this sampling step in an efficient manner with a complexity that naturally adapts to the rate of the posterior process. While it is hard to bound the required run-time in advance, tune the stopping criteria, or estimate the error of the approximation, this algorithm is the first to provide asymptotically unbiased samples for CTBNs.

A modern approach for developing state of the art inference algorithms for complex finite dimensional models that are faster than sampling is to use variational principles, where the posterior is approximated by a simpler and easier to manipulate distribution. To adopt this approach we show that candidate distributions can be parameterized by a set of continuous time-dependent functions. This representation allows us to develop a novel mean-field method, which approximates the posterior distribution using a product of independent inhomogeneous Markov processes. While this assumption introduces some bias, it results in a fast procedure. Moreover, it provides a lower bound on the likelihood of observations, which is important for learning procedures where we try to maximize the likelihood. This formulation results in an elegant algorithm that involves passing information about the posterior distribution in terms of continuous functions and processing these functions using sets of ordinary differential equations, which in turn are solved using highly optimized standard solvers. As such solvers can use an adaptive step size numerical integration, inference complexity is low for components which have uniform dynamics in some time segments.

The novel representation of posterior distributions presented here allows to consider variational approximations that are richer than mean-field thereby reducing the bias introduced by this algorithm. Specifically, we introduce a *belief propagation* algorithm that allows efficient inference while maintaining dependencies among different components by representing joint marginal distributions of sub-processes defined over overlapping clusters of components. Similarly to mean-field this results in a relatively simple algorithm which incorporates numerical integration of continuous functions. Empirical tests show that it leads to a significant improvement over mean-field providing highly accurate results on a range models.

Contents

Chapter 1

Introduction

Many real-life-tasks deal with complex dynamic systems that evolve in continuous-time. For example, robots evaluate their state using measurements form a continuously changing environment; Sensor networks evaluate fire hazard by measuring temperature and pressure in different rooms; Genomes of evolving species experience mutations that affect their structure and behavior. Such systems involve multiple components that affect each others' state through a network of interactions. Thus, a central challenge in these domains is to reason about continuous-time multi-component systems.

The goals of studying such systems are to understand what are the rules that govern the dynamics of the system as well as to predict the outcome of different scenarios. For example, a robotic-car is driving on a rocky terrain. We want to infer the probability that it will get stuck at different time points. The answer to this question depends both on the readings in different sensors as well as on the characteristics of the connections between its internal components.

A widely used approach to studying real world phenomena from measurements is to construct a suitable class of probabilistic models, and to search for a specific model that best fits the data [Koller and Friedman, 2009]. Applying this approach requires three fundamental ingredients: a mathematical *modeling language* that captures the essential characteristics of a process in a compact manner; a *learning procedure*, allowing to estimate the structure and parametrization of a specific model from observations; and an *inference procedure* allowing to perform predictions given a model and observations as well as to compute the likelihood of the observations. In this thesis, we deal with these three issues in the context of continuous-time multi-component dynamic systems.

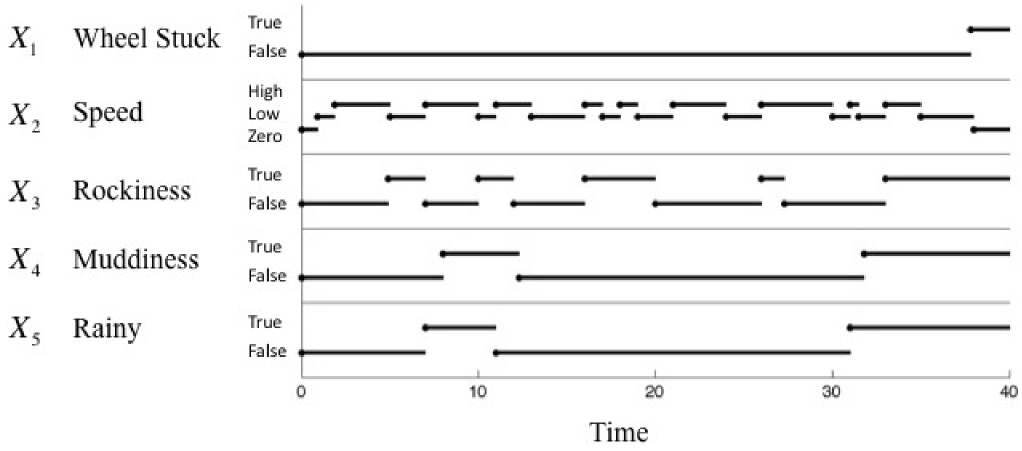


Figure 1.1: A trajectory of the rover monitoring system

1.1 Modeling Stochastic Systems in Continuous-Time

Probabilistic models of dynamic systems are called *stochastic processes*. Formally, a stochastic process is a collection of random variable $\{\boldsymbol{X}^{(t)}\}$ where t is a time index and each $\boldsymbol{X}^{(t)}$ assumes values from a set S termed the *state space* of the process. In other words, $\boldsymbol{X}^{(t)}$ describes the state of the system at time t. In a continuous-time process t belongs to a continuous segment of the real numbers. In this thesis we deal with multi-component processes where the random variables are vector valued $\boldsymbol{X}^{(t)} = (X_1^{(t)}, X_2^{(t)}, \ldots, X_D^{(t)})$ with a D-dimensional state-space $S = S_1 \times S_2 \times \ldots \times S_D$ where each S_i is finite.

For example, consider a simplified system for monitoring a robotic rover (adapted from Ng et al. [2005]). One of the system's role is to evaluate if one of the wheels gets stuck according to some sensor inputs. The system has four binary components whose state is one of {true,false} and indicating whether it is rainy, ground rockiness, ground muddiness and the state of the wheel. The system also monitors the speed of the robot (for simplicity let's assume is zero, low or high). A *trajectory* of a dynamic system describes the state of every component of the system at any time point. An example for such trajectory is shown in Figure 1.1. In this example the system starts in a non-rainy, non-muddy, non-rocky condition, the speed is zero and the wheel is not stuck. At $t = 0.92$ speed becomes low; at $t = 1.87$ speed becomes high; at $t = 4.9$ the terrain gets rocky; and so on. A stochastic process defines a distribution over such trajectories.

1.1.1 Continuous-Time Markov Processes

A common assumption taken in stochastic modeling is that when a system is described with sufficient detail its dynamics is *Markovian*; meaning that the future state is independent of the past given full knowledge of the present state. Therefore, Markov processes are ubiquitous in physics, chemistry, biology and technology [Gardiner, 2004].

A *continuous-time Markov process* (CTMP) is a stochastic process satisfying

$$\Pr(\boldsymbol{X}^{(t_{k+1})} = \boldsymbol{x}_{k+1}|\boldsymbol{X}^{(t_k)} = \boldsymbol{x}_k, \ldots, \boldsymbol{X}^{(t_1)} = \boldsymbol{x}_1) = \Pr(\boldsymbol{X}^{(t_{k+1})} = \boldsymbol{x}_{k+1}|\boldsymbol{X}^{(t_k)} = \boldsymbol{x}_k)$$

for every $t_1 < \ldots < t_k < t_{k+1}$ and states $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k, \boldsymbol{x}_{k+1}$. A CTMP is *time-homogeneous* if the conditional probability $\Pr(\boldsymbol{X}^{(t)} = \boldsymbol{y}|\boldsymbol{X}^{(s)} = \boldsymbol{x})$ depends on the time difference $t - s$ between the events regardless of the absolute time in which they took place.

When modeling a stochastic system, we want to capture its dynamics using a compact and intuitive parameterization. Such characterization should allow us to compute the probability of any finite number of point observations. The Markov assumption implies that a set of time-dependent conditional distributions is sufficient. Additionally, homogeneity allows to write these conditional distributions as time dependent transition matrices

$$p_{\boldsymbol{x},\boldsymbol{y}}(t) \equiv \Pr(\boldsymbol{X}^{(s+t)} = \boldsymbol{y}|\boldsymbol{X}^{(s)} = \boldsymbol{x}),$$

giving

$$\Pr(\boldsymbol{X}^{(t_k)} = \boldsymbol{x}_k, \ldots, \boldsymbol{X}^{(t_1)} = \boldsymbol{x}_1) = \Pr(\boldsymbol{X}^{(t_1)} = \boldsymbol{x}_1) \prod_{j=1}^{k-1} p_{\boldsymbol{x}_j,\boldsymbol{x}_{j+1}}(t_{j+1} - t_j) \ .$$

Therefore, the state distribution in a finite number of time points is characterized by an initial distribution and a time dependent matrix whose entries are $p_{\boldsymbol{x},\boldsymbol{y}}(t)$.

Under mild assumptions, the transition matrix have a simple parameterization. First, $p_{\boldsymbol{x},\boldsymbol{y}}(0) = \mathbf{1}_{\boldsymbol{x}=\boldsymbol{y}}$, where $\mathbf{1}$ is the indicator function. Next, assuming $p_{\boldsymbol{x},\boldsymbol{y}}(t)$ is continuous at $t = 0$ for every $\boldsymbol{x}$ and $\boldsymbol{y}$, these functions are also differentiable at $t = 0$. Their derivatives,

$$q_{\boldsymbol{x},\boldsymbol{y}} = \lim_{h \to 0} \frac{p_{\boldsymbol{x},\boldsymbol{y}}(h) - \mathbf{1}_{\boldsymbol{x}=\boldsymbol{y}}}{h},$$

are the entries of the *rate matrix* $\mathbb{Q}$. This equation implies that for small enough h and for $\boldsymbol{x} \neq \boldsymbol{y}$, the transition function satisfies $p_{\boldsymbol{x},\boldsymbol{y}}(h) \approx q_{\boldsymbol{x},\boldsymbol{y}}h$. Intuitively, higher transition rate from $\boldsymbol{x}$ to $\boldsymbol{y}$ implies higher probability of making this transition in a given time interval. By using this rate matrix it is possible to obtain the transition functions for any time interval with length t by computing

$$p_{\boldsymbol{x},\boldsymbol{y}}(t) = [\exp(\mathbb{Q}t)]_{\boldsymbol{x},\boldsymbol{y}}, \tag{1.1}$$

where $\exp(\mathbb{A})$ is the matrix exponential, defined by the Tailor series,

$$\exp(\mathbb{A}) = \mathbb{I} + \sum_{k=1}^{\infty} \frac{\mathbb{A}^k}{k!} \ .$$

Thus, the rate matrix $\mathbb{Q}$ characterizes the dynamics of the process (see Chapter 4 Section 2.1 for more details).

Continuous-time Markov processes have been thoroughly studied in the past century, giving rich and elegant mathematical foundations [Chung, 1960, Gardiner, 2004]. However, the problem of modeling such processes in large systems having many components is still in its early stages. The first challenge in that direction is constructing an appropriate modeling language. As the number of states is exponential in the number of components, a naive parameterization of a multi-component process requires a huge rate matrix.

1.1.2 Continuous-Time Bayesian Networks

To cope with the parameterization challenge we can make a reasonable assumption stating that in many applications interactions are sparse. In other words, the dynamics of every component is influenced by a small number of other components. *Continuous-time Bayesian networks* (CTBN) is an elegant modeling language suitable for such systems proposed by Nodelman et al. [2002]. A CTBN is composed of a directed graph where each node represents a component in the system and the edges represent direct influences between components. In this model it is assumed that only one component can change at a time, and transitions rates of a specific component depend only on its current state and on the state of its parents in the graph.

Formally, CTBNs are a subclass of multi-component CTMPs in which the rate matrix has a structure that corresponds to the graph structure. This CTMP is parameterized by *conditional rates* $q^{i|\mathbf{Pa}_i}_{x_i,y_i|\boldsymbol{u}_i}$, denoting the change rate of component i given the state $\boldsymbol{u}_i$ of its parents $\mathbf{Pa}_i$. The entries of the rate matrix of the multi-component

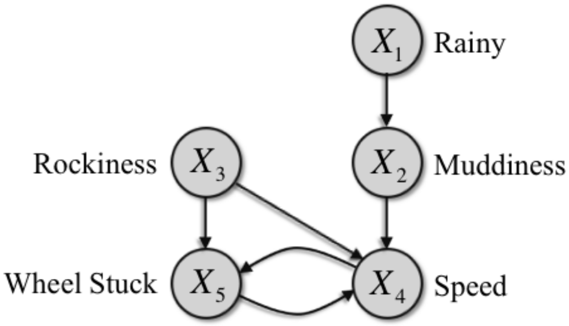


Figure 1.2: A CTBN describing a simplified robotic rover monitoring system

process are

$$q_{\boldsymbol{x},\boldsymbol{y}} = \begin{cases} q^{i|\mathbf{Pa}_i}_{x_i,y_i|\boldsymbol{u}_i} & \delta(\boldsymbol{x},\boldsymbol{y}) = \{i\} \\ \sum_i q^{i|\mathbf{Pa}_i}_{x_i,x_i|\boldsymbol{u}_i} & \boldsymbol{x} = \boldsymbol{y} \\ 0 & \text{otherwise,} \end{cases}$$

where $\delta(\boldsymbol{x},\boldsymbol{y}) = \{i|x_i \neq y_i\}$ is the number of components that differ between $\boldsymbol{x}$ and $\boldsymbol{y}$. Figure 1.2 presents a CTBN of the rover monitoring system. In this example, ground muddiness is only directly influenced by the rain condition. Note the cyclic dependency between the wheel-stuck and the speed. In this rover CTBN, the entry that corresponds to changing the state of X_2 from x_2 to y_2 while the rest of the system remains in state $X_i = x_i$ for every $i \neq 2$ is written as

$$q_{\{x_1,\underline{x_2},x_3,x_4,x_5\},\{x_1,\underline{y_2},x_3,x_4,x_5\}} = q^{1|2}_{x_2,y_2|x_1},$$

where $q^{1|2}_{x_2,y_2|x_1}$ is a conditional rate parameter of X_2 given the state x_1 of X_1.

This representation is both compact as well as providing a meaningful map of the interplay between different components. For example, as the rover system has $2 \times\ 2 \times 2 \times 2 \times 3 = 48$ states, a CTMP describing this system requires a full rate matrix with $48^2 = 2304$ parameters. On the other hand, the rover CTBN involves $2^2 + 2 \times 2^2 + 2^2 + 8 \times 3^2 + 6 \times 2^2 = 112$ parameters. A thorough introduction to CTBNs is provided in Chapter 4 Section 2.2.

1.1.3 Discretization Versus Continuous-Time Modeling

Before we begin addressing issues of continuous-time modeling, we should examine the alternative approach for using discrete-time models. A common approach to probabilistic modeling of dynamic systems is using *dynamic Bayesian networks* (DBNs), which are a compact representation for discrete-time Markov processes. To apply such models we discretize time into regular intervals of length h. A dynamic Bayesian network on the sampled time points $t_k = kh$ is composed of random variable set $\{X_i^{(t_k)}\}$ describing the state of component i at time t_k. The conditional distribution of $\boldsymbol{X}^{(t_{k+1})}$ given a specific state of $\boldsymbol{X}^{(t_k)}$ is modeled using a *directed acyclic graph* whose nodes represent random variable and edges represent direct probabilistic dependencies of nodes on their parents. The joint distribution of the process is parameterized by a set of local conditional probability tables for every variable $X_i^{(t_{k+1})}$ given the state of its parents in the graph. Therefore, the size of this representation depends on the sparseness of the graph. Although the inference and learning problems are intractable even for DBNs with sparse graphs, there are numerous works aiming at providing approximate solutions to these problems.

Using a discretized model for a continuous-time process have several limitations [Nodelman et al., 2003]. First, using a too small sampling interval h induces computational overhead while a coarse discretization looses information. Moreover, often different components evolve at different time scales and even the typical evolution rate of a single component can change across time. Therefore, there is no natural time scale suitable for discretizing the process.

Another important limitation is that while learning the topology of the graph of a DBN the resulting structure is dependent on the length of the time scale h. For a large enough time interval the state of a specific component depends not only on the subset that directly influence its dynamics, but also on additional components that influence this subset. The bigger h is, the larger the number of components that affect its state through such indirect interactions, resulting in a denser graph. This phenomenon is known as *entanglement*. Denser DBNs in turn require a larger number of parameters leading to less accurate statistical estimates. Additionally, they obscure the true structure of the process making it hard to distinguish between direct and indirect influences.

Reasoning on structured process directly in continuous-time can potentially lead to more accurate and succinct models (as demonstrated by Nodelman et al. [2003] in numerical experiments). Additionally, algorithms for continuous-time models can be more efficient, spending computational efforts in an adaptive manner. However, further research is required to bring this potential into practice.

1.2 Learning

Learning CTBNs is the process of constructing a specific model that describes empirical observations of the state of the system taken at different time points. This process involves both *parameter estimation* to evaluate the intensities of direct influences observed in the data, as well as *structure learning* performed by searching a graph that describes the structure of these influences. In this thesis, we assume the structure of the model is given and focus on the first problem.

The maximum likelihood approach to parameter estimation searches for a set of parameters that maximize the probability of the observed data. Applying this approach to CTMPs is straightforward given *fully observed data* . Such a data set is composed of trajectories of the process, which describe the state of the system at any time point (Figure 1.1). Each trajectory can be characterized by a finite sequence of states and transition times between them. In that case, the maximum likelihood estimator $\hat{q}_{\boldsymbol{x},\boldsymbol{y}}$ is a function of the observed residence time in state $\boldsymbol{x}$ denoted by $T_{\boldsymbol{x}}$; and the number of transitions from $\boldsymbol{x}$ to $\boldsymbol{y}$ denoted by $M_{\boldsymbol{x},\boldsymbol{y}}$. It is given by $\hat{q}_{\boldsymbol{x},\boldsymbol{y}} = \frac{M_{\boldsymbol{x},\boldsymbol{y}}}{T_{\boldsymbol{x}}}$. This equation has an appealing intuition: the best estimator for transition rates is the number of observed transitions divided by the residence time in the source state. Consequently, $T_{\boldsymbol{x}}$ and $M_{\boldsymbol{x},\boldsymbol{y}}$ are *sufficient statistics* for CTMPs, meaning that they summarize the data needed to compute the likelihood of the model.

Generalizing this property, the maximum likelihood estimator of a CTBN is also a function of sufficient statistics:

$$\hat{q}^{i|\mathbf{Pa}_i}_{x_i,y_i|\boldsymbol{u}_i} = \frac{M^{i|\mathbf{Pa}_i}_{x_i,y_i|\mathbf{Pa}_i}}{T^{i|\mathbf{Pa}_i}_{x_i|\boldsymbol{u}_i}} ,$$

where $T^{i|\mathbf{Pa}_i}_{x_i|\boldsymbol{u}_i}$ is the residence time of X_i in state x_i while his parents are in state $\boldsymbol{u}_i$, and $M^{i|\mathbf{Pa}_i}_{x_i,y_i|\mathbf{Pa}_i}$ is the number of transition of that X_i underwent from x_i to y_i while his parents were in state $\boldsymbol{u}_i$ [Nodelman et al., 2003]. Thus, in the case of fully observed trajectories parameter estimation of CTBN involves simple operations with sufficient statistics.

However, in typical applications only *partial observations* are available, reporting on the state of the system at specific time points or across short sub-intervals and possibly of only a subset of components. In that case, the sufficient statistics are unobserved. To handle this case, Nodelman et al. [2005a] proposed an iterative *expectation maximization* procedure for finding a local maximum of the likelihood function. This procedure starts by choosing some initial parameters. Then, on each iteration an *ex-*

pectation step computes the expected values of the required sufficient statistics given the partially observed data and the current model; a *maximization step* then maximize the estimators using the expected sufficient statistics as done in the fully observed case. The computationally challenging step is this procedure is the expectation step which requires to infer expected statistics in a CTBN.

1.3 Inference

Once we have a probabilistic model and partial evidence we can use it to reason about unseen events. Evidence may include complete point observations such as $\boldsymbol{X}^{(0)} = \boldsymbol{e}_0$, partial point observations such as $X_1^{(20)} = \mathsf{true}$ and $X_2^{(20)} = \mathsf{false}$ and interval observations such as $X_1^{([3.2,7])} = \mathsf{true}$, meaning that X_1 is true throughout the time interval $[3.2, 7]$. Inference is the task of answering queries about the posterior distribution given a series of such evidence. Possible queries include:

1. Posterior marginal distributions of some components $X_i^{(t)}$ in specific time points.
2. Expectations of statistics given the evidence, such as expected residence time or number of transitions of some components
3. The likelihood of the evidence

As showed in the previous section, in addition to its role in performing predictions, inference is a an important element of learning. Expectations are used for searching optimal parameters and the likelihood allows monitoring the progress of the learning process.

Since CTBNs are a subclass of CTMPs, exact inference can be performed in small models. For example, posterior probabilities are computed using time dependent transition probabilities $p_{\boldsymbol{x},\boldsymbol{y}}(t)$, which in turn can be computed by exponentiation of the rate matrix as described in Equation 1.1. However, the complexity of these operations scales with the size of the state-space which in turn is exponential in the number of components. Therefore, beyond a modest number of components we have to resort to approximations, which are the subject of chapters 3-5.

Approximate inference in multivariate models is a computationally challenging task. It is an active field of research rooted in physics (e.g. Metropolis et al. [1953]) and statistics [Gelman et al., 2003] and is still a hot research area in machine learning. Different algorithms provide different trade-off and are suited for different purposes. The inference problem is considered particularly hard in dynamic models, leading to specialized research and algorithms for discrete-time models [Murphy, 2002, Koller and Friedman, 2009] and recently for continuous-time models. Generally, the wealth of approximate inference algorithms falls into two main categories: sampling based [Gilks et al., 1996] and variational approximations [Wainwright and Jordan, 2008]. Here, we briefly describe these two approaches in the context of continuous-time models.

1.3.1 Sampling Based Approximations

Sampling-based inference is a randomized approach to estimating expectations of random variables. In the context of continuous-time processes sampling algorithms attempt to generate independent samples from the posterior distribution over trajectories such as those depicted in Figure 1.1. Such methods estimate the expectation of a random variable f by taking m samples $\sigma[1], \ldots, \sigma[m]$ from the posterior distribution and calculating the mean of these samples

$$\hat{\mathbf{E}}_{\sigma_{1:m}}[f] = \frac{1}{m}\sum_{i=1}^{m} f(\sigma[i]) \ .$$

For example, f can be a number of transitions, expected residence time or any other property of a given trajectory. Whenever $\sigma[i]$ are generated exactly from the posterior, this estimator is *unbiassed*, meaning that its expectation equals the estimated quantity. Moreover, if the samples are independent then the variance of this estimator is diminishes with $\frac{1}{m}$. In that case, as the number of sample increases the expected error decreases at a rate of $\frac{1}{\sqrt{m}}$.

While the tasks of sampling trajectories given evidence at $t = 0$ and of calculating the likelihood of a given sample are straightforward, sampling is more challenging for a general type of evidence. Recent works attempt to address this problem using a strategy called *importance sampling* in which samples are taken from a distribution different than the posterior and the contribution of each on of them is weighted according to their true posterior likelihood [Ng et al., 2005, Fan and Shelton, 2008, Fan et al., 2010]. These algorithms have an *anytime property* meaning that the error decreases as more samples are taken. However, in the case of low-probability point evidence the bias of the estimator decreases at a low rate and the procedure requires many samples. In Chapter 3 we present a sampling algorithm that works well regardless of the likelihood of the evidence.

1.3.2 Variational Algorithms

A modern approach for developing state of the art inference algorithms for complex finite dimensional models, which are generally faster than sampling, is to use variational principles. Variational algorithms approximate the posterior by simpler distributions. The specific approximation is chosen by searching the set of candidate distributions for the one that is closest to the posterior. A crucial point in applying this approach to continuous-time models is to define an appropriate representation of candidate distributions.

Nodelman et al. [2005b] applied this approach to CTBNs using a piecewise homogeneous representation over clusters of subsets of components. For each cluster $\mathcal{C}_i \subseteq \{1, \ldots, D\}$ the algorithm maintains a series of demarcation points $t_{i,1}, \ldots, t_{i,n_i}$ and a set of rate matrices one for each segment. Thus, the algorithm represents the posterior over each cluster as Markov process whose rates are constant within segments $(t_{i,k}, t_{i,k+1})$ but may change between these segments. The parameters of the rate matrices are updated using a *message passing scheme* which involves both update form a cluster $\mathcal{C}_i$ to $\mathcal{C}_j$ about the distribution over $\mathcal{C}_i \cap C_j$ as well as messages between consecutive segments within a cluster. This scheme is designed to search for rate parameters such that distributions over cluster are as close as possible to the posterior and such that pairs of clusters agree on the distribution over their intersection. Saria et al. [2007] improved this method by proposing an automatic procedure for choosing the boundaries of the homogeneous segments. This algorithm involves an elaborate message passing scheme which involves both updates of the parameterization as well as maintenance of the partition into segments. In Chapters 4 and 5 we present two variational algorithms that use an alternative representation of the posterior, which is both richer and simpler to handle.

1.3.3 Challenges in the Continuous-Time Domain

Approximate inference algorithms in discrete multivariate probabilistic have different tradeoffs. For example, some sampling algorithms are asymptotically unbiased while in many cases variational are very efficient. The dynamic nature of CTBNs and specifically the continuous-time domain impose additional specific issues: How should we sample continuous-time trajectories? How should we represent an approximation of the posterior in a manner that facilitates efficient computations while maintaining rich expressiveness? How do we handle processes whose components evolve in different time granularities that are unknown a-priory? On the other hand, as discussed in

Section 1.1.3, continuous-time modeling should have crucial advantages over potential disadvantages. Therefore, a major goal of this thesis is to address these issues in order to provide relatively simple and efficient algorithms by adopting principles from discrete models while exploiting the benefits of continuous-time modeling (Chapters 3-5).

1.4 Stationary Distributions of Structured Dynamic Processes

While CTBNs provide an intuitive modeling language for a wide range of domains, in some cases a model with different semantics is needed. An important example is evolution of living organisms which is driven by two opposing forces: *random mutations* in genomes of species and by *natural selection* . At the molecular level, these forces result in continuous change in DNA, RNA and protein sequences. The wealth genome sequences of different species, as well as structural and functional characterization of some molecules, enable an intensive study in the field of molecular evolution. Research on that direction requires dynamic models which enable us to learn the process in a manner that untwines selection constraints from random mutations.

To illustrate this type of dynamics let us examine the constraints acting on RNA molecules. An RNA molecule is a long chain of small molecules called nucleotides, each is one of four types denoted by A, C, G and U. The sequence of such a molecule determines the three dimensional structure to which it folds, which in turn determines its function. An essential factor that determines the structure on an RNA molecule is base-pairing, an interaction which involves either $\{\mathtt{A},\mathtt{U}\}$ pairs or $\{\mathtt{C},\mathtt{G}\}$ pairs. For example, in the RNA molecule depicted in Figure 1.3 the nucleotides in positions 30 and 37 are spatially adjacent as a result of such interactions. The stability of an RNA molecule depends also on *stacking interactions* between adjacent pairs such as between the pair in positions 30 and 37 and the pair positions 29 and 38. The network of such interactions and others induce constraints on the identity of nucleotides which should be maintained in an evolving molecule. The probability that a random mutation gets fixed in a population depends on the global fitness of the resulting molecule. To study evolution given a set of RNA sequences, we should be able to discern the effect of random mutations from the effect of the forces and interactions that determine fitness.

One approach to learning selection constraints is to attempt inferring the stationary

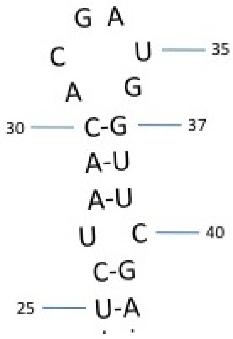


Figure 1.3: Secondary structure of a subsequence of an RNA molecule.

distribution of the process

$$\boldsymbol{\pi}_{\boldsymbol{x}} = \lim_{t\to\infty} \Pr(\boldsymbol{X}^{(t)} = \boldsymbol{x} | \boldsymbol{X}^{(0)} = \boldsymbol{y}) \ ,$$

which under reasonable assumptions is unique, meaning that it does not depend on the initial state $\boldsymbol{y}$. The premise of this approach is that sequences $\boldsymbol{x}$ satisfying the selection constraints are likely to survive within populations in the long term. Therefore, the stationary distribution assigns high probability to such sequences.

As the number of possible sequences is exponential, a naive representation of the stationary distribution is impractical and not informative. Hence, a useful representation of this distribution should have a more compact form. Importantly, such representation should be learnable from temporal data, where samples are taken across different time points and are not independent. In Chapter 2 we address this problem by introducing a novel continuous-time modeling language, in which there is a direct relation between the representation of transient dynamics and the stationary distribution.

1.5 Related Models and Applications

Several continuous-time learnable models were applied in several domains since the introduction of CTBNs . Different models can be classified according to the structure of the state-space, the nature of influences between components and the observation model.

One category of models include processes with discrete state spaces. Fan and Shelton [2009] presented an application of CTBNs to modeling the dynamics of social networks. Gopalratnam et al. [2005] introduced an extended CTBN model in which the distribution of a component's transition time given its parents is richer than the one of CTBN. Rajaram et al. [2005] introduced Poisson-network where each component is a Poisson process whose rate depends on its parents in the graph. In this model each component is a counting process, meaning that it can take any non-negative integer value. Another model with conditional Poisson processes yet with a different dependency structure was introduced by Simma et al. [2008] and was applied to modeling information traffic in computer networks.

A continuous-time Markov model with discrete state space but with noisy observations was presented by Opper and Sanguinetti [2007]. In this model the hidden components are counting processes representing the number of species an ecological application and number of molecules in a molecular-biology application.

Another category of models includes both discrete and continuous valued components. Ng et al. [2005] introduced a state estimation model of an experimental NASA Mars rover. The discrete components dynamics is modeled by a CTBN, whereas continuous-time components follow nonlinear noisy dynamics that depends on the state of the discrete ones. Continuous time components are observed with some measurement noise. Hybrid models were also used to analyze gene expression dynamics in cells using a simple system involving one or two stochastic elements [Sanguinetti et al., 2009, Opper and Sanguinetti, 2010].

Purely continuous-state models with intra-component interactions can be represented as stochastic differential equations. Archambeau et al. [2007] presented an approximate inference method for such models.

Additional models include spatio-temporal processes, which describe the dynamics of system of particles that diffuse through space, and can participate in chemical reactions. A potential application demonstrated for this model is analyzing the dynamics of substances affecting the development of a fruit fly embryo [Ruttor and Opper, 2010, Dewar et al., 2010]. Finally, continuous-time topic models represent the evolution of distributions of words given each topic [Wang et al., 2008].

The above works handle the inference and learning problems using approximations that are suitable for these specific models. In this thesis we focus on structured discrete-state processes with noiseless evidence.

1.6 Research Goals and Thesis Outline

This thesis deals with the three elements required to reason about the continuous-time dynamic of structured systems: representation, learning and inference. We begin by addressing the first two elements (Chapter 2). Then we shall devote a major part of this thesis to deal with the computationally intensive inference problem. We develop three different approximations, each one with different tradeoffs, aiming to provide a set of tools suitable for different stages in the study of dynamic systems (Chpaters 3-5).

Chapter 2 presents a new modeling language providing an explicit representation of the stationary distribution which is compact and learnable from temporal data [El-Hay et al., 2006]. Our motivating application is modeling co-evolution of interacting elements in bio-sequences, an intensively studied problem in the field of molecular evolution. However, the proposed model is general, as it is equivalent to a large subclass of processes that can be modeled by CTBNs.

The goal of our efforts on the inference problem is to achieve high accuracy, efficiency and simplicity. Our strategy is to use solid theoretical foundations of inference in discrete models, combined with well studied algorithms for handling continuous time-dependent functions, such as adaptive numerical integration.

In Chapter 3 we present an inference algorithm based on a Gibbs sampling strategy [El-Hay et al., 2008]. This procedure - rather than sampling a fixed number of random variables at a time - samples complete trajectories. Similarly to its discrete counterpart, the resulting estimates converge to the true ones, but convergence rates may be slow and hard to monitor.

Chapter 4 introduces an intuitive representation of posterior distributions in terms of continuous functions, and use a simplified version of this representation to develop a mean-field algorithm [Cohn et al., 2010]. This algorithm is not only fast but also is the only one that computes a lower bound on the probability of evidence. Although it is biased, it provides good results in the context of molecular-evolution.

Finally, in Chapter 5 we introduce a *belief propagation* algorithm that is based on similar principles developed for the mean-field algorithm, yet uses a richer representation for the posterior [El-Hay et al., 2010]. Whereas the mean field algorithm is faster and has convergence guaranties, this algorithm gives highly accurate results in a broad set of regimes.

Chapter 2

Paper: Continuous-Time Markov Networks

Tal El-Hay, Nir Friedman, Daphne Koller, and Raz Kupferman

In *Proceedings of the Twenty-second Conference on Uncertainty in AI (UAI)*, 2006

Continuous Time Markov Networks

Tal El-Hay **Nir Friedman**
School of Computer Science
The Hebrew University
{tale,nir}@cs.huji.ac.il

Daphne Koller
Department of Computer Science
Stanford University
koller@cs.stanford.edu

Raz Kupferman
Institute of Mathematics
The Hebrew University
raz@math.huji.ac.il

Abstract

A central task in many applications is reasoning about processes that change over continuous time. Recently, Nodelman et al. introduced *continuous time Bayesian networks (CTBNs)*, a structured representation for representing *Continuous Time Markov Processes* over a structured state space. In this paper, we introduce *continuous time Markov networks (CTMNs)*, an alternative representation language that represents a different type of continuous-time dynamics, particularly appropriate for modeling biological and chemical systems. In this language, the dynamics of the process is described as an interplay between two forces: the tendency of each entity to change its state, which we model using a continuous-time *proposal process* that suggests possible local changes to the state of the system at different rates; and a global *fitness* or *energy* function of the entire system, governing the probability that a proposed change is accepted, which we capture by a Markov network that encodes the fitness of different states. We show that the fitness distribution is also the stationary distribution of the Markov process, so that this representation provides a characterization of a temporal process whose stationary distribution has a compact graphical representation. We describe the semantics of the representation, its basic properties, and how it compares to CTBNs. We also provide an algorithm for learning such models from data, and demonstrate its potential benefit over other learning approaches.

1 Introduction

In many applications, we reason about processes that evolve over time. Such processes can involve short time scales (e.g., the dynamics of molecules) or very long ones (e.g., evolution). In both examples, there is no obvious discrete "time unit" by which the process evolves. Rather, it is more natural to view the process as changing in a continuous time: the system is in some state for a certain duration, and then transitions to another state. The language of *continuous time Markov processes* (CTMPs) provides an elegant mathematical framework to reason about the probability of trajectories of such systems. Unfortunately, when we consider a system with multiple components, this representation grows exponentially in the number of components. Thus, we aim to construct a representation language for CTMPs that can compactly encode natural processes over high-dimensional state spaces. Importantly, the representation should also facilitate effective inference and learning.

Recently, Nodelman et al. [8, 9, 10, 11] introduced the representation language of *continuous time Bayesian networks* (CTBNs), which provides a factorized, component-based representation of CTMP: each component is characterized by a conditional CTMP dynamics, which describes its local evolution as a function of the current state of its parents in the network. This representation is natural for describing systems with a sparse structure of local influences between components. Nodelman et al. provide algorithms for efficient approximate inference in CTBNs, and for learning them from both complete and incomplete data.

In this paper, we introduce *continuous time Markov networks*, which have a different representational bias. Our motivating example is modeling the evolution of biological sequences such as proteins. In this example, the state of the system at any given time is the sequence of amino acids encoded by the gene of interest. As evolution progresses, the sequence is continually modified by local mutations that change individual amino acids. The mutations for different amino acids occur independently, but the probability that these local mutations survive depends on global aspects of the new sequence. For example, a mutation may be accepted only if the new sequence of amino acids folds properly into a functional protein, which occurs only if pairs of amino acids that are in contact with each other in the folded protein have complementary charges. Thus, although the modifications are local, global constraints on the protein structure and function introduce dependencies.

To capture such situations, we introduce a representation where we specify the dynamics of the process using two components. The first is a *proposal process* that attempts to change individual components of the system. In our example, this process will determine the rate of random mutations in protein sequences. The second is an equilibrium distribution, which encodes preferences over global configurations of the system. In our example, an approximation of the fitness of the folded protein. The equilibrium distribution is a static quantity that encodes preferences among states of the system, rather than dynamics of changes. The actual dynamics of the system are determined by the interplay between these two forces: local mutations and global fitness. We represent the equilibrium distribution compactly using a Markov network, or, more generally, a feature-based log-linear model.

Importantly, as we shall see, the equilibrium distribution parameter is indeed the *equilibrium* distribution of the process. Thus, our representation provides an explicit representation of both the dynamics of the system and its asymptotic limit. Moreover, this representation ensures that the equilibrium distribution has a pre-specified simple structure. Thus, we can view our framework as a *continuous-time Markov network* (CTMN) — a Markov network that evolves over continuous time. From a different perspective, our representation allows us to capture a family of temporal processes whose stationary distribution has a certain locality structure. Such processes occur often in biological and physical systems. For example, recent results of Socolich et al. [13] suggest that pairwise Markov networks can fairly accurately capture the fitness of protein sequences.

We provide a reduction from CTMNs to CTBNs, allowing us to use CTBN algorithms [7, 11] to perform effective approximate inference in CTMNs. More importantly, we also provide a procedure for learning CTMN parameters from data. This procedure allows us to estimate the stationary distribution from observations of the system's dynamics. This is important in applications where the stationary distribution provides insight about the domain of application. In the protein evolution example, the stationary distribution provides a description of the evolutionary forces that shape the protein and thus gives important clues about protein structure and function.

2 Reversible Continuous Time Markov Processes

We now briefly summarize the relevant properties of continuous time Markov processes that will be needed below. We refer the interested reader to Taylor and Karlin [14] and Chung [2] for more thorough expositions. Suppose we have a family of random variables $\{\boldsymbol{X}(t) : t \geq 0\}$ where the continuous index t denotes time. A joint distribution over these random variables is a homogeneous *continuous time Markov process* (CTMP) if it satisfies the *Markov property*

$$\Pr(\boldsymbol{X}(t_{k+1})|\boldsymbol{X}(t_k), \ldots, \boldsymbol{X}(t_0)) = \Pr(\boldsymbol{X}(t_{k+1})|\boldsymbol{X}(t_k))$$

for all $t_{k+1} > t_k > \ldots > t_0$, and time-homogeneity,

$$\Pr(\boldsymbol{X}(s+t) = \boldsymbol{y}|\boldsymbol{X}(s) = \boldsymbol{x}) = \\ \Pr(\boldsymbol{X}(s'+t) = \boldsymbol{y}|\boldsymbol{X}(s') = \boldsymbol{x})$$

for all s, s' and $t > 0$.

The dynamics of a CTMP are fully determined by the *Markov transition function*,

$$p_{\boldsymbol{x},\boldsymbol{y}}(t) = \Pr(\boldsymbol{X}(s+t) = \boldsymbol{y}|\boldsymbol{X}(s) = \boldsymbol{x}),$$

where time-homogeneity implies that the right hand side does not depend on s. Provided that the transition function satisfies certain analytical properties (see [2]) the dynamics are fully captured by a constant matrix $\boldsymbol{Q}$ — the *rate*, or *intensity matrix* — whose entries $q_{\boldsymbol{x},\boldsymbol{y}}$ are defined by

$$q_{\boldsymbol{x},\boldsymbol{y}} = \lim_{h \downarrow 0} \frac{p_{\boldsymbol{x},\boldsymbol{y}}(h) - \boldsymbol{1}\{\boldsymbol{x} = \boldsymbol{y}\}}{h}, \tag{1}$$

where $\boldsymbol{1}\{\}$ is the *indicator function* which takes the value 1 when the condition in the argument holds and 0 otherwise. The Markov process can also be viewed as a generative process: The process starts in some state $\boldsymbol{x}$. After spending a finite amount of time at $\boldsymbol{x}$, it transitions, at a random time, to a random state $\boldsymbol{y} \neq \boldsymbol{x}$. The transition times to the various states are exponentially distributed, with rate parameters $q_{\boldsymbol{x},\boldsymbol{y}}$. The diagonal elements of $\boldsymbol{Q}$ are set to ensure the constraint that each row sums up to zero.

If the process satisfies certain conditions (reachability) then the limit

$$\boldsymbol{\pi}_{\boldsymbol{x}} = \lim_{t \to \infty} p_{\boldsymbol{y},\boldsymbol{x}}(t)$$

exists and is independent of the initial state $\boldsymbol{y}$. That is, in the long time limit, the probability of visiting state $\boldsymbol{x}$ is independent of the initial state at time 0. The distribution $\boldsymbol{\pi}_{\boldsymbol{x}}$ is called the *stationary distribution* of the process. A CTMP is called *stationary* if $P(\boldsymbol{X}(0) = \boldsymbol{x}) = \boldsymbol{\pi}_{\boldsymbol{x}}$, that is, if the initial state is sampled from the stationary distribution. A stationary CTMP is called *reversible* if for every $\boldsymbol{x}, \boldsymbol{y}$, and $t > 0$

$$\Pr(X(t) = \boldsymbol{y}|X(0) = \boldsymbol{x}) = \Pr(X(0) = \boldsymbol{y}|X(t) = \boldsymbol{x}).$$

This condition implies that the process is statistically equivalent to itself running backward in time. Reversibility is intrinsic to many physical systems where the microscopic dynamics are time-reversible. Reversibility can be formulated as a property on the Markov transition function, where for every $\boldsymbol{x}, \boldsymbol{y}$, and $t > 0$

$$\boldsymbol{\pi}_{\boldsymbol{x}} p_{\boldsymbol{x},\boldsymbol{y}}(t) = \boldsymbol{\pi}_{\boldsymbol{y}} p_{\boldsymbol{y},\boldsymbol{x}}(t).$$

This identity is known as the *detailed balance* condition. To better understand the constraint, we can examine the implications of reversibility on the rate matrix $\boldsymbol{Q}$.

Proposition 2.1: *A CTMP is reversible if and only if its rate matrix can be expressed as*

$$q_{\boldsymbol{x},\boldsymbol{y}} = \boldsymbol{\pi}_{\boldsymbol{y}} s_{\boldsymbol{x},\boldsymbol{y}},$$

where $s_{\boldsymbol{x},\boldsymbol{y}}$ are the entries of a symmetric matrix (that is, $s_{\boldsymbol{x},\boldsymbol{y}} = s_{\boldsymbol{y},\boldsymbol{x}}$).

In other words, in a reversible CTMP, the asymmetry in transition rates can be interpreted as resulting entirely from preferences of the stationary distribution.

3 Continuous Time Metropolis Processes

We start by considering a reformulation of reversible CTMPs as a continuous time version of the Metropolis sampling process. We view the process as an interplay between two factors. The first is an unbiased random process that attempts to transition between states of the system, and the second is the tendency of the system to remain in more probable states. This latter probability is taken to be the stationary distribution of the process. The structure of the process can be thought of as going through iterations of proposed transitions that are either accepted or rejected, similar to the Metropolis sampler [6].

To formally describe such a process, we need to describe these two components. The first is the unbiased proposal of transitions. These proposals occur at fixed rates. We denote by $r_{\boldsymbol{x},\boldsymbol{y}}$ the rate at which proposals to transition $\boldsymbol{x} \to \boldsymbol{y}$ occur. This in effect defines a CTMP process with rate matrix $\boldsymbol{R}$. To ensure an unbiased proposal, we require $\boldsymbol{R}$ to be symmetric. (The stationary distribution of a symmetric rate matrix is the uniform distribution.)

The second component is a decision whether to accept or reject the proposed transition. The decision whether to accept the transition $\boldsymbol{x} \to \boldsymbol{y}$ depends on the probability ratio of these states at equilibrium. We assume that we are given a target distribution, which should coincide with the equilibrium distribution $\boldsymbol{\pi}$. As we shall see, to reach the target equilibrium distribution, the acceptance probability should satisfy a simple condition. To make this precise, we assume we have an *acceptance function* f that takes as an argument the ratio $\boldsymbol{\pi}_{\boldsymbol{y}}/\boldsymbol{\pi}_{\boldsymbol{x}}$ and returns the probability of accepting transition $\boldsymbol{x} \to \boldsymbol{y}$. This function should return a value between 0 and 1, and satisfy the functional relation

$$f(z) = zf\left(\frac{1}{z}\right). \tag{2}$$

Two functions that satisfy these conditions are

$$\begin{aligned} f_{\text{Metropolis}}(z) &= \min(1, z) \\ f_{\text{logistic}}(z) &= \frac{1}{1+\frac{1}{z}}. \end{aligned}$$

The function $f_{\text{Metropolis}}$ is the standard one used in Metropolis sampling. The function f_{logistic} is closely linked to logistic regression. It is continuously differentiable, which, as we shall see, facilitates the subsequent analysis.

Formally, a *continuous time Metropolis process* is defined by a symmetric matrix $\boldsymbol{R}$, a distribution $\boldsymbol{\pi}$, and a real-valued function f. The semantics of the process are defined in a generative manner. Starting at an initial state $\boldsymbol{x}$, the system remains in the state until receiving a proposed transition $\boldsymbol{x} \to \boldsymbol{y}$ with rate $r_{\boldsymbol{x},\boldsymbol{y}}$. This proposal is then accepted with probability $f(\boldsymbol{\pi}_{\boldsymbol{y}}/\boldsymbol{\pi}_{\boldsymbol{x}})$. If it is accepted, the system transitions to state $\boldsymbol{y}$; otherwise it remains in state $\boldsymbol{x}$. This process is repeated indefinitely.

To formulate the statistical dynamics of the system, consider a short time interval h. In this case, the probability of a proposal of the transition $\boldsymbol{x} \to \boldsymbol{y}$ is roughly $h \cdot r_{\boldsymbol{x},\boldsymbol{y}}$. Since the proposed transition is accepted with probability $f(\boldsymbol{\pi}_{\boldsymbol{y}}/\boldsymbol{\pi}_{\boldsymbol{x}})$, we have:

$$p_{\boldsymbol{x},\boldsymbol{y}}(h) \approx h \cdot r_{\boldsymbol{x},\boldsymbol{y}} \cdot f\left(\frac{\boldsymbol{\pi}_{\boldsymbol{y}}}{\boldsymbol{\pi}_{\boldsymbol{x}}}\right).$$

Plugging this into Eq. (1) we conclude that the off-diagonal elements of $\boldsymbol{Q}$ are

$$q_{\boldsymbol{x},\boldsymbol{y}} = r_{\boldsymbol{x},\boldsymbol{y}} \cdot f\left(\frac{\boldsymbol{\pi}_{\boldsymbol{y}}}{\boldsymbol{\pi}_{\boldsymbol{x}}}\right). \tag{3}$$

Proposition 3.1: *Consider a continuous time Metropolis process defined as in Eq. (3). Then, this CTMP is reversible, and its stationary distribution is $\boldsymbol{\pi}$.*

Proof: The condition on f implies that

$$\frac{1}{\boldsymbol{\pi}_{\boldsymbol{y}}} f\left(\frac{\boldsymbol{\pi}_{\boldsymbol{y}}}{\boldsymbol{\pi}_{\boldsymbol{x}}}\right) = \frac{1}{\boldsymbol{\pi}_{\boldsymbol{x}}} f\left(\frac{\boldsymbol{\pi}_{\boldsymbol{x}}}{\boldsymbol{\pi}_{\boldsymbol{y}}}\right),$$

Thus, it follows that $q_{\boldsymbol{x},\boldsymbol{y}}$ is of the form $s_{\boldsymbol{x},\boldsymbol{y}}\boldsymbol{\pi}_{\boldsymbol{y}}$, i.e., that the process is reversible. Moreover, it implies that the stationary distribution of the process is $\boldsymbol{\pi}$. ▮

The inverse result is also easy to obtain.

Proposition 3.2: *Any reversible CTMP can be represented as a continuous time Metropolis process.*

Proof: According to Proposition 2.1 we can write $q_{\boldsymbol{x},\boldsymbol{y}} = \boldsymbol{\pi}_{\boldsymbol{y}} s_{\boldsymbol{x},\boldsymbol{y}}$ for a symmetric matrix $s_{\boldsymbol{x},\boldsymbol{y}}$. Define

$$r_{\boldsymbol{x},\boldsymbol{y}} = s_{\boldsymbol{x},\boldsymbol{y}} \frac{\boldsymbol{\pi}_{\boldsymbol{y}}}{f\left(\frac{\boldsymbol{\pi}_{\boldsymbol{y}}}{\boldsymbol{\pi}_{\boldsymbol{x}}}\right)},$$

so that $q_{\boldsymbol{x},\boldsymbol{y}} = r_{\boldsymbol{x},\boldsymbol{y}} \cdot f\left(\frac{\boldsymbol{\pi}_{\boldsymbol{y}}}{\boldsymbol{\pi}_{\boldsymbol{x}}}\right)$. Together, $s_{\boldsymbol{x},\boldsymbol{y}} = s_{\boldsymbol{y},\boldsymbol{x}}$ and Eq. (2) imply that $r_{\boldsymbol{x},\boldsymbol{y}} = r_{\boldsymbol{y},\boldsymbol{x}}$. Thus, $\boldsymbol{R}$ is symmetric and together with $\boldsymbol{\pi}$ defines a continuous time Metropolis process which is equivalent to the original reversible CTMP. ▮

We conclude that continuous time Metropolis processes are a general reparameterization of reversible CTMPs.

4 Continuous Time Markov Networks

We are interested in dealing with structured, multi-component systems, whose state description can be viewed as an assignment to some set of state variables $\boldsymbol{X} = \langle X_1, X_2, \ldots, X_n\rangle$, where each X_i assumes a finite set of values. The main challenge is dealing with the large state space (exponential in n). We aim to find succinct representations of the system's dynamics within the framework of continuous time Metropolis processes. We do so in two stages, first dealing with the proposal rate matrix $\boldsymbol{R}$, and then with the equilibrium distribution $\boldsymbol{\pi}$.

Our first assumption is that proposed transitions are local. Specifically, we require that, for $\boldsymbol{x} \neq \boldsymbol{y}$

$$r_{\boldsymbol{x},\boldsymbol{y}} = \begin{cases} r^i_{x_i,y_i} & (x_j = y_j)\ \forall j \neq i \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where $\boldsymbol{R}^i = \{r^i_{x_i,y_i}\}$ are symmetric local transition rates for X_i. Thus, we allow only one component to change at a time and the proposal rates do not depend on the global state of the system.

The second assumption concerns the structure of the stationary distribution $\boldsymbol{\pi}$. *Log-linear models* or *Markov networks* provide a general framework to describe structured distributions. A log-linear model is described by a set of *features*, each one encoding a local property of the system that involves few variables. For example, the function $\mathbb{1}\{X_1 = X_2\}$ is a feature that only involves two variables.

A feature-based Markov network is defined by a vector of features, $\boldsymbol{s} = \langle s_1, \ldots, s_K\rangle$, where each feature s_k assigns a real number to the state of the system. We further assume that each feature s_k is a function of a (usually small) subset $\boldsymbol{D}_k \subseteq \boldsymbol{X}$ of variables. We use the notation $\boldsymbol{x}|_{\boldsymbol{D}_k}$ to denote the projection of $\boldsymbol{x}$ on the subset of variables $\boldsymbol{D}_k$. Thus, s_k is a function of $\boldsymbol{x}|_{\boldsymbol{D}_k}$; however, for notational convenience, we sometimes use $s_k(\boldsymbol{x})$ as a shorthand for $s_k(\boldsymbol{x}|_{\boldsymbol{D}_k})$.

Based on a set of features, we define a distribution by assigning different weights to each feature. These weights represent the relative importance of each feature. We use the notation $\boldsymbol{\theta} = \langle \theta_1, \ldots, \theta_K\rangle \in \mathbb{R}^K$ to denote the vector of weights or *parameters*. The equilibrium distribution represented by $\boldsymbol{s}$ and $\boldsymbol{\theta}$ takes the log-linear form

$$\boldsymbol{\pi}_{\boldsymbol{x}} = \frac{1}{Z(\boldsymbol{\theta})} \exp\left\{\sum_k \theta_k \cdot s_k(\boldsymbol{x}|_{\boldsymbol{D}_k})\right\}, \tag{5}$$

where the *partition function* $Z(\boldsymbol{\theta})$ is the normalizing factor.

The structure of the equilibrium distribution can be represented as an undirected graph $\mathcal{G}$ — the nodes of $\mathcal{G}$ represent the variables $\{X_1, \ldots, X_n\}$. If $X_i, X_j \in \boldsymbol{D}_k$ for some k, then there is an edge between the corresponding nodes. Thus, for every feature s_k, the nodes that represent the variables in $\boldsymbol{D}_k$ form a clique in the graph $\mathcal{G}$. We define the *Markov Blanket*, $\mathcal{N}_{\mathcal{G}}(i)$, of the variable X_i as the set of neighbors of X_i in the graph $\mathcal{G}$ [12].

Example 4.1 : Consider a four-variable process $\{X_1, X_2, X_3, X_4\}$, where each variable takes binary values, with the following set of features:

$$\begin{aligned} s_1(X_1) &= \mathbb{1}\{X_1 = 1\} & s_5(X_1, X_2) &= \mathbb{1}\{X_1 = X_2\} \\ s_2(X_2) &= \mathbb{1}\{X_2 = 1\} & s_6(X_2, X_3) &= \mathbb{1}\{X_2 = X_3\} \\ s_3(X_3) &= \mathbb{1}\{X_3 = 1\} & s_7(X_3, X_4) &= \mathbb{1}\{X_3 = X_4\} \\ s_4(X_4) &= \mathbb{1}\{X_4 = 1\} & s_8(X_1, X_4) &= \mathbb{1}\{X_1 = X_4\} \end{aligned}$$

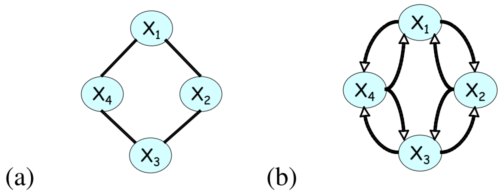


Figure 1: (a) The Markov network structure for Example 4.1. (b) The corresponding CTBN structure.

Note that all these features involve at most two variables. The corresponding graph structure is shown in Figure 1(a). In this example $\mathcal{N}(1) = \{X_2, X_4\}$, $\mathcal{N}(2) = \{X_1, X_3\}$, $\mathcal{N}(3) = \{X_2, X_4\}$, and $\mathcal{N}(4) = \{X_1, X_3\}$. ▮

We now take advantage of the structured representation of both $\boldsymbol{R}$ and $\boldsymbol{\pi}$ to get a more succinct representation of the rate matrix $\boldsymbol{Q}$ of the process. We exploit the facts that $\boldsymbol{\pi}$ appears explicitly in the rate only as a ratio $\boldsymbol{\pi}_{\boldsymbol{y}}/\boldsymbol{\pi}_{\boldsymbol{x}}$, and moreover that the proposal process includes only transitions that modify a single variable. Thus, we only examine ratios where $\boldsymbol{y}$ and $\boldsymbol{x}$ agree on all variables but one. It is straightforward to show that if $\boldsymbol{x}$ and $\boldsymbol{y}$ are two states that are identical except for the value of X_i and $\boldsymbol{u}_i = \boldsymbol{x}|_{\mathcal{N}(i)}$, then

$$\boldsymbol{\pi}_{\boldsymbol{y}}/\boldsymbol{\pi}_{\boldsymbol{x}} = g_i(x_i \to y_i|\boldsymbol{u}_i),$$

where

$$g_i(x_i \to y_i|\boldsymbol{u}_i) = \exp\left\{\sum_{k:X_i\in\boldsymbol{D}_k} \theta_k[s_k(y_i, \boldsymbol{u}_i) - s_k(x_i, \boldsymbol{u}_i)]\right\}.$$

Note that, if $X_i \in \boldsymbol{D}_k$, then $\boldsymbol{D}_k \subseteq \mathcal{N}(i) \cup \{X_i\}$. Thus, the function g_i is well defined.

Thus, the acceptance probability of a change in X_i depends only on the state of variables in its Markov blanket. This property is heavily used for Gibbs sampling in Markov networks. Depending on the choice of features, these dependencies can be very sparse, or involve all the variables in the process.

To summarize, assuming a local form for $\boldsymbol{R}$ and a log-linear form for $\boldsymbol{\pi}$, we can further simplify the definition of the rate matrix $\boldsymbol{Q}$. If $\boldsymbol{x}$ and $\boldsymbol{y}$ are two states that differ only in the i'th variable, then

$$q_{\boldsymbol{x},\boldsymbol{y}} = r^i_{x_i,y_i} f(g_i(x_i \to y_i|\boldsymbol{u}_i)), \tag{6}$$

where $\boldsymbol{u}_i = \boldsymbol{x}|_{\mathcal{N}(i)}$. All other off-diagonal entries are 0, and the diagonal entries are set to ensure that the sum of each row is 0. We call a process with a $\boldsymbol{Q}$ matrix of the form Eq. (6) a *Continuous time Markov Network* (CTMN).

One consequence of the form of the CTMN rate matrix Eq. (6) is that the dynamics of the i'th variable depend directly only on the dynamics of its neighbors. As we can expect, we can use this property to discuss independencies among variables in the network. However, since we are examining a continuous process, we need to consider independencies between full trajectories (see also [8]).

Theorem 4.2: Consider a CTMN with a stationary distribution represented by a graph $\mathcal{G}$. If $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$ are subsets of $\boldsymbol{X}$ such that $\boldsymbol{C}$ separates $\boldsymbol{A}$ from $\boldsymbol{B}$ in $\mathcal{G}$, then the trajectories of $\boldsymbol{A}$ and $\boldsymbol{B}$ are conditionally independent, given observation of the full trajectory of $\boldsymbol{C}$.

Proof: (sketch) Using the global independence properties of a Markov network (see for example, [12]), we have that $\boldsymbol{\pi}$ can be written as a product of two function each with its own domain $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ such that $\boldsymbol{X}_1 \cap \boldsymbol{X}_2 = \boldsymbol{C}$ and $\boldsymbol{A} \subseteq \boldsymbol{X}_1$ and $\boldsymbol{B} \subseteq \boldsymbol{X}_2$. Once the trajectories of variables in $\boldsymbol{C}$ are given, the dynamics of variables in $\boldsymbol{X}_1 - \boldsymbol{C}$ and $\boldsymbol{X}_2 - \boldsymbol{C}$ are two independent CTMNs, each with its own stationary distribution. As a consequence we get the desired independence. ▮

That is, the usual conditional separation criterion in Markov networks [12] applies in a trajectory-wise fashion to CTMNs.

It is important to note that although we can represent any reversible CTMP as a continuous time Metropolis process, once we move to CTMNs this is no longer the case. The main restriction is that, in CTMNs as we have defined them, each transition involves a change in the state of exactly one component. Thus, although the language of Markov networks allow to describe arbitrary equilibrium distributions (potentially with an exponential number of features), the restrictions on $\boldsymbol{R}$ limit the range of processes we can describe as CTMNs. As an example of a domain where CTMNs are not suitable, consider reasoning about biochemical systems, where each component of the state is the number of molecules of a particular species and transitions correspond to chemical reactions. For example, a reaction might be one that takes an OH molecule and an H molecule and replace them by an H_2O molecule. If reactions are reversible (i.e., H_2O can break into OH and H molecules), then this process might be described by a reversible CTMP. However, since reactions change several components at once, we cannot describe such system as a CTMN.

5 Connection to CTBNs

The factored form of Eq. (6) allows us to relate CTMNs with CTBNs. A CTBN is defined by a *directed* (often cyclic) graph whose nodes correspond to variables of the process, and whose edges represent direct influences of one variable on the evolution of another. More precisely, a CTBN is defined by a collection of *conditional rate matrices* (also called conditional intensity matrices). For each X_i, and for each possible value $\boldsymbol{u}_i$ of its direct parents in the CTBN graph, the matrix $\boldsymbol{Q}^{X_i|\boldsymbol{u}_i}$ is a rate matrix over the state space of X_i. These conditional rate matrices are combined into a global rate matrix by a process Nodelman et al. [9] call amalgamation. Briefly, if $\boldsymbol{x}$ and $\boldsymbol{y}$ are identical except for the value of X_i, then

$$q_{\boldsymbol{x},\boldsymbol{y}} = q^{X_i|\boldsymbol{u}_i}_{x_i,y_i} \tag{7}$$

where $\boldsymbol{u}_i = \boldsymbol{x}|_{\mathbf{Pa}_i}$ is the assignment to X_i's parents in the state $\boldsymbol{x}$. That is, the rate of transition from $\boldsymbol{x}$ to $\boldsymbol{y}$ is the conditional rate of X_i changing from x_i to y_i given the state of its parents. Again, all other off-diagonal elements, where more than one variable changes, are set to 0.

This form is similar to the rate matrix of CTMNs shown in Eq. (6). Thus, given a CTMN, we can build an equivalent CTBN by setting the parents of each X_i to be $\mathcal{N}(i)$, and using the conditional rates:

$$q^{X_i|\boldsymbol{u}_i}_{x_i,y_i} = r^i_{x_i,y_i} g_i(x_i \to y_i \mid \boldsymbol{x}|_{\mathcal{N}(i)}) \tag{8}$$

Figure 1(b) shows the CTBN structures corresponding to the CTMN of Example 4.1. In general, the CTBN graph corresponding to a given CTMN is built by replacing each undirected arc by a pair of directed ones. This matches the intuition that if X_i and X_j appear in the context of some feature, then they mutually influence each other.

As this transformation shows, the class of processes that can be encoded using CTMNs is a subclass of CTBNs. In a sense, this is not surprising, as a CTBN can encode any Markov process where at most one variable can transition at a time. However, the CTMN representation imposes a particular parametrization of the system dynamics in terms of the local proposal process and the global equilibrium distribution. This parametrization violates both local and global parameter independence [5] in the resulting CTBN. In particular, a transition between x_i and y_i is proposed at the same rate, regardless of whether it is globally advantageous (in terms of equilibrium preferences). As we shall see, this property is important for our ability to effectively estimate these rate parameters.

Moreover, as we have seen, this parametrization guarantees that the stationary distribution of the process factorizes as a particular Markov network. In general, even a fairly sparse CTBN gives rise to a fully entangled stationary distribution that cannot be factorized. Indeed, even computing the stationary distribution of a given CTBN is a hard computational problem. By contrast, we have defined a model of temporal dynamics that gives rise to a natural and interpretable form for the stationary distribution. This property is critical in applications where the stationary distribution is the key element in understanding the system.

Yet, the ability to transform a CTMN into a CTBN allows us to harness the recently developed approximate inference methods for CTBNs [11, 7], including for the E-step used when learning CTMNs for partially observable data.

6 Parameter Learning

We now consider the problem of learning the parametrization of CTMNs from data. Thus, we assume we are given the form of $\boldsymbol{\pi}$, that is, the set of features $\boldsymbol{s}$, and need to learn the parameters $\boldsymbol{\theta}$ governing $\boldsymbol{\pi}$ and the local rate matrices $\boldsymbol{R}^i$ that govern the proposal rates for each variable. We start by considering this problem in the context of *complete data*, where our observations consist of full trajectories of the system. As we show, we define a gradient ascent procedure for learning the parameters from such data.

This result also enables us to learn from incomplete data using the standard EM procedure. Namely, we can use existing CTBNs inference algorithms to perform the E-step effectively when learning from partially observable data to compute expected sufficient statistics. The M-step is then an application of the learning procedure for complete data with these expected sufficient statistics. This combination is quite standard and follows the lines of similar procedure for CTBNs [10], and therefore we do not expand on it here.

6.1 The Likelihood Function

A key concept in addressing the learning problem is the likelihood function, which determines how the probability of the observations depends on the parameters.

We assume that the data is complete, and thus our observations consist of a trajectory of the system that can be described as a sequence of intervals, where in each interval the system is in one state. Using the relationship to CTBNs, we can use the results of Nodelman *et al.* [9] to write the probability of the data as a function of sufficient statistics and entries in the conditional rate matrices of Eq. (8). A problem with this approach is that the entries in the conditional rate matrix involve both parameters from $\boldsymbol{R}^i$ and parameters from $\boldsymbol{\theta}$. Thus, the resulting likelihood function couples the estimation of these two sets of parameters.

However, if we had additional information, we could decouple these two sets of parameters. Suppose we observe not only the actual trajectories, but also the rejected proposals; see Figure 2. With this additional information, we can estimate the rate of different proposals, independently of whether they were accepted or not. Similarly, we can estimate the equilibrium distribution from the accepted and rejected proposals. Thus, we are going to view our learning problem as a partial data problem where the annotation of rejected proposals is the missing data.

To formalize these ideas, assume that our evidence is a trajectory annotated with proposal attempts. We describe such a trajectory using three vectors; see Figure 2. The first vector, $\boldsymbol{\tau} = \langle \tau[1], \ldots, \tau[M+1] \rangle$, represents the time intervals between consecutive proposals. Thus, the first proposal took place at time $\tau[1]$, the second at time $\tau[1]+\tau[2]$, and so on. The last entry in this vector is the time between the last proposal and the end of the observed time interval. The second vector, $\Xi = \langle \boldsymbol{x}[0], \boldsymbol{x}[1], \ldots, \boldsymbol{x}[M] \rangle$, denotes the actual state of the system after each proposal was made. Thus, $\boldsymbol{x}[0]$ is the initial state of the system, $\boldsymbol{x}[1]$ is the state after the first proposal, and so on. Finally, $\Upsilon = \langle \boldsymbol{y}[1], \ldots, \boldsymbol{y}[M] \rangle$ denotes the sequence of proposed states. Clearly, the m'th proposal was accepted if $\boldsymbol{y}[m] = \boldsymbol{x}[m]$ and rejected otherwise. We denote these event using the indicators $S[m] = \boldsymbol{1}\{\boldsymbol{x}[m] = \boldsymbol{y}[m]\}$.

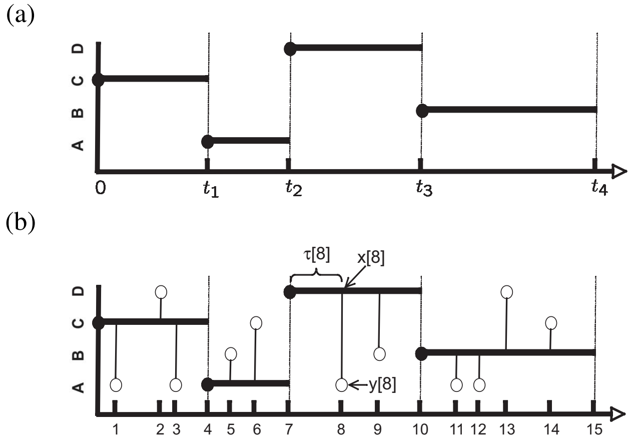


Figure 2: An illustration of training data. (a) A complete trajectory. The x-axis denotes time and the y-axis denotes the state at each time. Filled circles denote transitions. (b) A trajectory annotated with accepted and rejected proposals (closed and open circles, respectively). (Remember that accepted proposals lead to a transition.) The marks on the x-axis denote the index of the proposal. We illustrate the notation we use in the text, where $\tau[i]$ denotes the time interval before the i'th proposal, $\boldsymbol{x}[i]$ denote the actual state after the i'th proposal, and $\boldsymbol{y}[i]$ denote the proposed state in the i'th proposal.

The likelihood of these observations is the product of the probability density of the duration between proposals, and the probability of accepting or rejecting each proposal. Plugging in the factored form of $\boldsymbol{R}$ and $\boldsymbol{\pi}$ we can write this likelihood in a compact form.

Proposition 6.1: *Given an augmented data set, $\boldsymbol{\tau}$, Ξ, and Υ, the log-likelihood can be decomposed as*

$$\ell(\boldsymbol{\theta}, \{\boldsymbol{R}^i\} : \boldsymbol{\tau}, \Xi, \Upsilon) = \sum_{i=1}^{n} \ell_{r,i}(\boldsymbol{R}^i : \boldsymbol{\tau}) + \ell_{\boldsymbol{s}}(\boldsymbol{\theta} : \Xi, \Upsilon),$$

such that

$$\ell_{r,i}(\boldsymbol{R}^i : \boldsymbol{\tau}) = \sum_{x_i \neq y_i} \left(M\left[x_i, y_i\right] \ln r^i_{x_i,y_i} - r^i_{x_i,y_i} T\left[x_i\right] \right)$$

and

$$\ell_{\boldsymbol{s}}(\boldsymbol{\theta} : \Xi, \Upsilon) =$$
$$\sum_{i=1}^{n}\sum_{\boldsymbol{u}_i}\sum_{x_i \neq y_i} M^a\left[x_i, y_i|\boldsymbol{u}_i\right] \ln f(g_i(x_i \to y_i|\boldsymbol{u}_i)) +$$
$$\sum_{i=1}^{n}\sum_{\boldsymbol{u}_i}\sum_{x_i \neq y_i} M^r\left[x_i, y_i|\boldsymbol{u}_i\right] \ln(1 - f(g_i(x_i \to y_i|\boldsymbol{u}_i)))$$

where $M^a\left[x_i, y_i|\boldsymbol{u}_i\right]$ is the number of accepted transitions of X_i from x_i to y_i when $\mathcal{N}(i) = \boldsymbol{u}_i$, $M^r\left[x_i, y_i|\boldsymbol{u}_i\right]$ is the count of rejected proposals to make the same transition, $M\left[x_i, y_i|\boldsymbol{u}_i\right] = M^a\left[x_i, y_i|\boldsymbol{u}_i\right] + M^r\left[x_i, y_i|\boldsymbol{u}_i\right]$, and $T\left[x_i\right]$ is the time spent in states where $X_i = x_i$.

Note that if we use f_{logistic}, then, as $\ln((1 + e^{-x})^{-1})$ is concave, the likelihood function $\ell_{\boldsymbol{s}}(\boldsymbol{\theta} : \Xi, \Upsilon)$ is concave and has a unique maximum.

6.2 Maximizing the Likelihood Function

Under the Maximum Likelihood Principle, our estimated parameters are the ones that maximize the likelihood function given the observations. We now examine how to maximize the likelihood. The decoupling of the likelihood into several terms allows us to estimate each set of parameters separately.

The estimation of $\boldsymbol{R}^i$ is straightforward: imposing the symmetry condition, the maximum likelihood estimate is

$$r^i_{x_i,y_i} = \frac{M\left[x_i, y_i\right] + M\left[y_i, x_i\right]}{T\left[x_i\right] + T\left[y_i\right]}.$$

Finding the maximum likelihood parameters of $\boldsymbol{\pi}$ is somewhat more involved. Note that the likelihood $\ell_{\boldsymbol{s}}(\boldsymbol{\theta} : \Xi, \Upsilon)$ is quite different from the likelihood of a log-linear distribution given i.i.d. data [3]. The probability of acceptance or rejection involves ratios of probabilities. Therefore, the partition function $Z(\boldsymbol{\theta})$ cancels out, and does not appear in the likelihood.

In a sense, our likelihood is closely related to the *pseudo-likelihood* for log-linear models [1]. Recall that pseudo-likelihood is a technique for estimating the parameters of a Markov network (or log-linear model) that uses a different objective function. Rather than optimizing the joint likelihood, one optimizes a sum of log conditional likelihood terms, one for each variable given its neighbors. By considering the conditional probability of a variable given its neighbors, the partition function cancels out, allowing the parameters to be estimated without the use of inference. At the large sample limit, optimizing the pseudo-likelihood criterion is equivalent to optimizing the joint likelihood, but the results for finite sample sizes tend to be worse. In our setting, the generative model is defined in terms of ratios only. Thus, in this case the exact likelihood turns out to take a form similar to the pseudo-likelihood criterion. As for pseudo-likelihood, this form allows us to perform parameter estimation without requiring inference in the underlying Markov network.

In the absence of an analytical solution for this equation we learn the parameters using a gradient-based optimization procedure to find a (local) maximum of the likelihood. The derivation of the gradient is a standard exercise; for completeness, we provide the details in the appendix. When using f_{logistic} we are guaranteed that such a procedure finds the unique global maximum.

6.3 Completing the Data

Our derivation of the likelihood and the associated optimization procedure relies on the assumption that rejected transition attempts are also observed in the data. As we can see from the form of the likelihood, these failures play an important role in estimating the parameters. The question is how to adapt the procedure to the case where rejected proposals are not observed. Our solution to this problem is to use Expectation Maximization, where we view the proposal attempts as the unobserved variables.

In this approach, we start with an initial guess of the model parameters. We use these to estimate the expected number of rejected proposals; we then treat these expected counts as though they were real, and maximize the likelihood using the procedure described in the previous section. We repeat these iterations until convergence.

The question is how to compute the expected number of rejected attempts. It turns out that this computation can be done analytically.

Proposition 6.2: *Given a CTMN, and an observed trajectory $\boldsymbol{\tau}, \Xi$. Then,*

$$\begin{aligned}&\boldsymbol{E}[M^r\left[x_i, y_i|\boldsymbol{u}_i\right]|\mathcal{D}] \\ &\quad= T\left[x_i|\boldsymbol{u}_i\right] r^i_{x_i,y_i}\left(1 - f(g(x_i, y_i|\boldsymbol{u}_i))\right)\end{aligned} \tag{9}$$

where $T\left[x_i|\boldsymbol{u}_i\right]$ is the total amount of time the system was in states where $X_i = x_i$ and $\mathcal{N}(i) = \boldsymbol{u}_i$.

We see that, in this case, the E-step of EM is fairly straightforward. The harder step is the M-step which requires an iterative gradient-based optimization procedure.

To summarize the procedure, to learn from complete data we perform the following steps: We first collect sufficient statistics $T\left[x_i|\boldsymbol{u}_i\right]$ and $M^a\left[x_i, y_i|\boldsymbol{u}_i\right]$. We then initialize the model with some set of parameters (randomly, or using prior knowledge). We then iterate over the two steps of EM until convergence: in the E-step, we complete the sufficient statistics with the expected number of rejected attempts, as per Eq. (9); in the M-step, we perform maximum likelihood estimation using the expected sufficient statistics, using gradient descent with the gradient of Eq. (10).

7 A Numerical Example

To illustrate the properties of our CTMN learning procedure, we evaluated it on a small synthetic data set. We

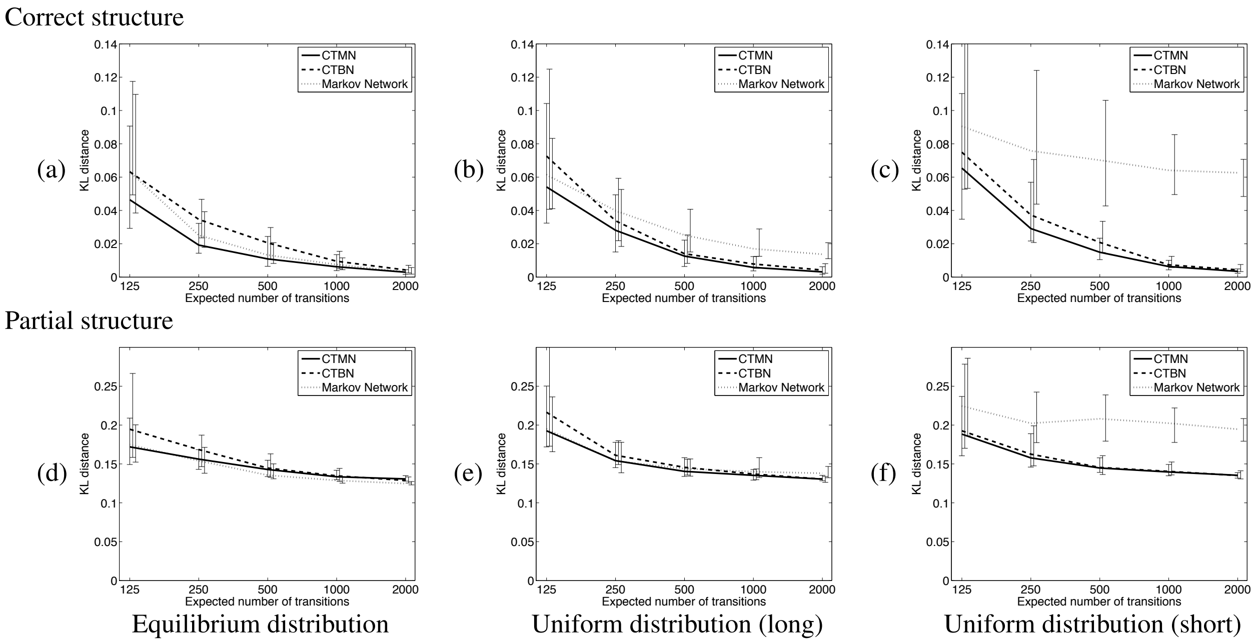


Figure 3: Comparison of estimates of the equilibrium distribution by the CTMN learning procedure (solid lines), the CTBN learning procedure (dashed lines) and a Markov Network parameter learning procedure applied to the frequency of time spent in each state (dotted lines). The x-axis denotes the total length of training trajectories (measured in units of expected number of observed transitions). The y-axis denotes the KL-divergence between the equilibrium distribution of the true model and the estimated model. The curves report the median performance among 50 data sets, and the error bars report $25\% - 75\%$ percentiles. (a-c) report performance when learning with the true structure from which the data was generated, and (d-f) report results when learning the parameters of a structure without the edges between X_1 and X_4. In (a) and (d) $p(\boldsymbol{X}(0))$ is the equilibrium distribution. In (b) and (e) $p(\boldsymbol{X}(0))$ is uniform and each trajectory is of length 25 time units. In (c) and (f) $p(\boldsymbol{X}(0))$ is uniform and each trajectory is of length 10 time units.

generated data from the CTMN model of Example 4.1 with $\boldsymbol{\theta} = \langle -0.2, -2.3, 0.7, 0.7, -1.2, -1.2, -1.2, -1.2 \rangle$ and proposal rates $r^1_{0,1} = 1$, $r^2_{0,1} = 2$, $r^3_{0,1} = 3$, and $r^4_{0,1} = 4$.

The goal of our experiments is to test the ability of the CTMN learning procedure to estimate stationary distributions from data in various conditions. As a benchmark, we compared our procedure to two alternative methods:

- A procedure that estimates the stationary distribution directly from the frequency of visiting each state. This procedure is essentially the standard parameter learning method for Markov networks, where the weight of each state (instance) is proportional to the duration in which the process spends in that state. This procedure uses gradient ascent to maximize the likelihood [3]. When the process is sampling from the stationary distribution, the relative time in each state is proportional to its stationary probability, and in such situations we expect this procedure to perform well.

- A procedure that estimates the $\boldsymbol{Q}$-matrix of the associated CTBN shown in Figure. 1. Here we used the methods designed for parameter learning of CTBNs in [9]. Once that the $\boldsymbol{Q}$-matrix has been estimated, the estimated stationary distribution is the only normalized vector in its null space.

We examined these three procedures in three sets of synthetic trajectories. The first set was generated by sampling the initial state $\boldsymbol{X}(0)$ of each trajectory from the stationary distribution and then sampling further states and durations from the target model. In this data set the system is in equilibrium throughout the trajectory. The second data set was generated by sampling the initial state from a uniform distribution, and so the system starts in a distribution that is far from equilibrium. However, the trajectory is long enough to let the system equilibrate. The third data set is similar to the second, except that trajectories are shorter and thus do not have sufficient time to equilibrate. To evaluate the effect of training set size, we repeated the learning experiments with different numbers of trajectories. We report the size of the training set in terms of the total length of training trajectories. Time is reported in units of *expected transition number*. That is, one time unit is equal to the average time between transitions when the process is in equilibrium. The short and long trajectories in our experiments are of length 10 and 25 expected transitions, respectively.

To evaluate the quality of the learned distribution, we measured the Kullback-Leibler divergences from the true stationary distribution to the estimated ones. Figures 3(a-c) show the results of these experiments. When sampling from the stationary distribution, the three procedures tend, as the data size increases, toward the correct distribution. For small data size, the performance of the CTMN learning procedure is consistently superior, although the error bars partially overlap. We start seeing a difference between the estimation procedures when we modify the initial distribution. As expected, the Markov network learning procedure suffers since it is learning from a biased sample. On the other hand, the performance of the CTMN and CTBN learning procedures is virtually unchanged, even when we modify the length of the trajectories. These results illustrate the ability of the CTMN and CTBN learning procedures to robustly estimate the equilibrium distribution from the dynamics even when the sampled process is not at equilibrium.

To test the robustness to the network structure, we also tested the performance of these procedures when estimating using a wrong structure. As we can see in Figures 3(d-f), while the three procedures converge to the wrong distribution, their relative behavior remains similar to the previous experiment, and the performance of the CTMN learning procedure is still not affected by the nature of the data.

8 Discussion and Future Work

In this paper, we define the framework of continuous time Markov networks, where we model a dynamical system as being governed by two factors: a local transition model, and a global acceptance/rejection model (based on an equilibrium distribution). By using a Markov network (or feature-based log-linear model) to encode the equilibrium distribution, we naturally define a temporal process guaranteed to have an equilibrium distribution of a particular, factored form. We showed a reduction from CTMNs to CTBNs that illustrates the differences in the expressive powers of the two formalisms. Moreover, this reduction allows us to reason in CTMNs by exploiting the efficient approximate inference algorithms for CTBNs. Finally, we provided learning algorithms for CTMNs, which allow us to learn the equilibrium distribution in a way that exploits our understanding about the system dynamics. We demonstrated on that this learning procedure is able to robustly estimate the equilibrium distribution even when the sampled process is not at equilibrium. These results can be combined for learning from partial observations, by plugging in the learning procedure as the M-step in the EM procedure for CTBNs [10].

This work opens many interesting questions. A key goal in learning these models is to estimate the stationary distribution. It is interesting to analyze, both theoretically and empirically, the benefit gained in this task by accounting for the process dynamics, as compared to learning the stationary distribution directly from a set of snapshots of the system (e.g., a set of instances of a protein sequence in different species). Moreover, so far, we have tackled only the problem of parameter estimation in these models. In many applications, the model structure is unknown, and of great interest. For example, in models of protein evolution, we want to know which pair of positions in the protein are directly correlated, and therefore likely to be structurally interacting. Of course, tackling this problem involves learning the structure of a Markov network, a notoriously difficult task. From the perspective of inference, our reduction to CTBNs can lose much of the structure of the model. For example, if the stationary distribution is a pairwise Markov network, the fact that the interaction model decomposes over pairs of variables is lost in the induced CTBN. It is interesting to see whether one can construct inference algorithms that better exploit this structure. Finally, one important limitation of the CTMN framework is the restriction to an exponential distribution on the duration between proposed state changes. Although such a model is a reasonable one in many systems (e.g., biological sequence evolution), there are other settings where it is too restrictive. In recent work, Nodelman et al. [10] show how one can expand the framework of CTBNs to allow a richer set of duration distributions. Essentially, their solution introduces a "hidden state" internal to a variable, so that the overall transition model of the variable is actually the aggregate of multiple transitions of its internal state. A similar solution can be applied in our setting, but the resulting model would not generally encode a reversible CTMP.

One major potential field of application for this class of models is sequence evolution. The current state of the art in phylogenetic inference is based on continuous time probabilistic models of evolution [4]. Virtually all of these models assume that sequence positions evolve independently of each other (although in some models, there are global parameters that induce weak dependencies). Our models provide a natural language for modeling such dependencies. In this domain, the proposal process corresponds to mutation rates within the sequence, and the equilibrium distribution is proportional to the relative fitness of different sequences. The latter function is of course very complex, but there is empirical evidence that modeling pairwise interactions can provide a good approximation [13]. Thus, in these systems, both the local mutation process and a factored equilibrium distribution are very appropriate, making CTMNs a potentially valuable tool for modeling and analysis. We hope to incorporate this formalism within phylogenetic inference tools, and to develop a methodology to leverage these models to provide new insights about the structure and function of proteins.

Acknowledgments

We thank A. Jaimovich, T. Kaplan, M. Ninio, I. Wiener, and the anonymous reviewers for comments on earlier versions of this manuscript. This work was supported by grants from the Israel Science Foundation and the US-Israel Binational Science Foundation, and by DARPA's CALO program, under sub-contract to SRI International.

A Gradient for Learning CTMNs

We now compute the derivative of the gradient of the log-likelihood, as specified in Proposition 6.1. The parameters $\boldsymbol{\theta}$ appear within the scope of the g_i functions. Thus, to find the derivatives we differentiate these functions with respect to the parameters, and then apply the chain rule for derivatives:

$$\begin{aligned}&\frac{\partial}{\partial\theta_k}\ell_{\boldsymbol{s}}(\boldsymbol{\theta}:\Xi,\Upsilon) = \\ &\quad\sum_{i:X_i\in\boldsymbol{D}_k}\sum_{\boldsymbol{u}_i}\sum_{x_i\neq y_i} \\ &\qquad\Delta_k(x_i,y_i|\boldsymbol{u}_i)\left(\psi_a(x_i,y_i|\boldsymbol{u}_i)M^a\left[x_i,y_i|\boldsymbol{u}_i\right] -\right. \\ &\qquad\qquad\left.\psi_r(x_i,y_i|\boldsymbol{u}_i)M^r\left[x_i,y_i|\boldsymbol{u}_i\right]\right)\end{aligned} \tag{10}$$

where

$$\begin{aligned}\Delta_k(x_i,y_i|\boldsymbol{u}_i) &= s_k(\boldsymbol{u}_i,y_i) - s_k(\boldsymbol{u}_i,x_i) \\ \psi_a(x_i,y_i|\boldsymbol{u}_i) &= \left.\frac{zf'(z)}{f(z)}\right|_{z=g_i(x_i\to y_i|\boldsymbol{u}_i)} \\ \psi_r(x_i,y_i|\boldsymbol{u}_i) &= \left.\frac{zf'(z)}{1-f(z)}\right|_{z=g_i(x_i\to y_i|\boldsymbol{u}_i)}\end{aligned}$$

This shows that the update of θ_k is a weighted combination of the contribution of each proposed transition. The weight of the transition depends on how sensitive the ratio of probabilities is to the feature, denoted by $\Delta_k(x_i,y_i|\boldsymbol{u}_i)$ and the number of times this transition was accepted or rejected, captured by the empirical counts. In addition, each proposal is weighted by $\psi_a(x_i,y_i|\boldsymbol{u}_i)$, which captures the improbability of the acceptance (respectively rejection for $\psi_r(x_i,y_i|\boldsymbol{u}_i)$). The less probable they are, the larger the change in θ_k.

We can get better understanding of these terms if we consider their value for specific choices of f. For example, if we use f_{logistic}, then

$$\begin{aligned}\psi_a(x_i,y_i|\boldsymbol{u}_i) &= 1 - f_{\text{logistic}}(g_i(x_i\to y_i|\boldsymbol{u}_i)) \\ \psi_r(x_i,y_i|\boldsymbol{u}_i) &= f_{\text{logistic}}(g_i(x_i\to y_i|\boldsymbol{u}_i)),\end{aligned}$$

that is, the rejection and acceptance probabilities, respectively. The smaller these values, the more probable was the acceptance (resp. rejection) and so it results in a smaller gradient of the likelihood in the direction of this parameter. When using $f_{\text{Metropolis}}$ the two functions are not symmetric:

$$\begin{aligned}\psi_a(x_i,y_i|\boldsymbol{u}_i) &= \boldsymbol{1}\{g_i(x_i\to y_i|\boldsymbol{u}_i)<1\} \\ \psi_r(x_i,y_i|\boldsymbol{u}_i) &= \\ &\boldsymbol{1}\{g_i(x_i\to y_i|\boldsymbol{u}_i)>1\}f_{\text{logistic}}(g_i(x_i\to y_i|\boldsymbol{u}_i))\end{aligned}$$

with a discontinuity when $g_i(x_i\to y_i|\boldsymbol{u}_i)=1$. We see that, in this case, the updates are asymmetric, with maximal weight to updates of accepted transitions.

References

[1] J. Besag. On the statistical analysis of dirty pictures. *J. Roy. Stat. Soc. B Met.*, 48(3):259–302, 1986.

[2] K.L. Chung. *Markov chains with stationary transition probabilities*. 1960.

[3] S. Della Pietra, V. Della Pietra, and J. Lafferty. Inducing features of random fields. *IEEE Trans. PAMI*, 19(4):380–393, 1997.

[4] J. Felsenstein. *Inferring Phylogenies*. 2004.

[5] D. Heckerman, D. Geiger, and D. M. Chickering. Learning Bayesian networks: The combination of knowledge and statistical data. *Mach. Learn.*, 20:197–243, 1995.

[6] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller. Equation of state calculation by fast computing machines. *J. Chem. Phys.*, 21:1087–1092, 1953.

[7] B. Ng, A. Pfeffer, and R. Dearden. Continuous time particle filtering. In *IJCAI '05*. 2005.

[8] U. Nodelman, C.R. Shelton, and D. Koller. Continuous time Bayesian networks. In *UAI '02*, pp. 378–387. 2002.

[9] U. Nodelman, C.R. Shelton, and D. Koller. Learning continuous time Bayesian networks. In *UAI '03*, pp. 451–458. 2003.

[10] U. Nodelman, C.R. Shelton, and D. Koller. Expectation maximization and complex duration distributions for continuous time Bayesian networks. In *UAI '05*, pp. 421–430. 2005.

[11] U. Nodelman, C.R. Shelton, and D. Koller. Expectation propagation for continuous time Bayesian networks. In *UAI '05*, pp. 431–440. 2005.

[12] J. Pearl. *Probabilistic Reasoning in Intelligent Systems*. 1988.

[13] M. Socolich, *et al.* Evolutionary information for specifying a protein fold. *Nature*, 437(7058):512–518, 2005.

[14] H.M. Taylor and S. Karlin. *An Introduction to Stochastic Modeling*. 1998.

Chapter 3

Paper: Gibbs sampling in factorized continuous-time Markov processes

Tal El-Hay, Nir Friedman, and Raz Kupferman

In *Proceedings of the Twenty-Fourth Conference on Uncertainty in AI (UAI), 2008*

Gibbs Sampling in Factorized Continuous-Time Markov Processes

Tal El-Hay **Nir Friedman**
School of Computer Science
The Hebrew University
{tale,nir}@cs.huji.ac.il

Raz Kupferman
Institute of Mathematics
The Hebrew University
raz@math.huji.ac.il

Abstract

A central task in many applications is reasoning about processes that change over continuous time. *Continuous-Time Bayesian Networks* is a general compact representation language for multi-component continuous-time processes. However, exact inference in such processes is exponential in the number of components, and thus infeasible for most models of interest. Here we develop a novel Gibbs sampling procedure for multi-component processes. This procedure iteratively samples a trajectory for one of the components given the remaining ones. We show how to perform *exact* sampling that adapts to the natural time scale of the sampled process. Moreover, we show that this sampling procedure naturally exploits the structure of the network to reduce the computational cost of each step. This procedure is the first that can provide asymptotically unbiased approximation in such processes.

1 Introduction

In many applications, we reason about processes that evolve over time. Such processes can involve short time scales (e.g., the dynamics of molecules) or very long ones (e.g., evolution). In both examples, there is no obvious discrete "time unit" by which the process evolves. Rather, it is more natural to view the process as changing in a continuous time: the system is in some state for a certain duration, and then transitions to another state. The language of *continuous-time Markov processes* (CTMPs) provides an elegant mathematical framework to reason about the probability of trajectories of such systems (Gardiner, 2004). We consider Markov processes that are homogeneous in time and have a finite state space. Such systems are fully determined by the state space S, the distribution of the process at the initial time, and a description of the dynamics of the process. These dynamics are specified by a *rate matrix* $\mathbb{Q}$, whose off-diagonal entries $q_{\boldsymbol{a},\boldsymbol{b}}$ are exponential rate intensities for transitioning from state $\boldsymbol{a}$ to $\boldsymbol{b}$. Intuitively, we can think of the entry $q_{\boldsymbol{a},\boldsymbol{b}}$ as the rate parameter of an exponential distribution whose value is the duration of time spent in state $\boldsymbol{a}$ before transitioning to $\boldsymbol{b}$.

In many applications, the state space is of the form of a product space $S = S_1 \times S_1 \times \cdots \times S_M$, where M is the number of *components* (such processes are called multi-component). Even if each of the S_i is of low dimension, the dimension of the state space is exponential in the number of components, which poses representational and computational difficulties. Recently, Nodelman et al. (2002) introduced the representation language of *continuous-time Bayesian networks* (CTBNs), which provides a factorized, component-based representation of CTMPs: each component is characterized by a conditional CTMP dynamics, which describes its local evolution as a function of the current state of its parents in the network. This representation is natural for describing systems with a sparse structure of local influences between components.

For most applications of such CTMP models, we need to perform inference to evaluate the posterior probability of various queries given evidence. Exact inference requires exponentiation of the rate matrix $\mathbb{Q}$. As the rate matrix is exponential in the number of components, exact computations are infeasible for more than a few components. Thus, applications of factored CTMPs require the use of approximate inference.

In two recent works Nodelman et al. (2005) and Saria et al. (2007) describe approximate inference procedures based on Expectation Propagation, a variational approximation method (Minka, 2001; Heskes and Zoeter, 2002). These approximation procedures perform local propagation of messages between components (or sub-trajectories of components) until convergence. Such procedures can be quite efficient, however they can also introduce a systematic error in the approximation (Fan and Shelton, 2008).

More recently, Fan and Shelton (2008) introduced a procedure that employs importance sampling and particle filtering to sample trajectories from the network. Such a stochastic sampling procedure has anytime properties as collecting more samples leads to more accurate approximation. However, since this is an importance sampler, it has limited capabilities to propagate evidence "back" to influence the sampling of earlier time steps. As a result, when the evidence is mostly at the end of the relevant time interval, and is of low probability, the procedure requires many samples. A related importance sampler was proposed by Ng et al. (2005) for monitoring a continuous time process.

In this paper we introduce a new stochastic sampling procedure for factored CTMPs. The goal is to sample random system trajectories from the posterior distribution. Once we have multiple independent samples from this distribution we can approximate the answer to queries about the posterior using the empirical distribution of the samples. The challenge is to sample from the posterior. While generative sampling of a CTMP is straightforward, sampling given evidence is far from trivial, as evidence modifies the posterior probability of earlier time points.

Markov Chain Monte Carlo (MCMC) procedures circumvent this problem by sampling a stochastic sequence of system states (trajectories in our models) that will eventually be governed by the desired posterior distribution. Here we develop a Gibbs sampling procedure for factored CTMPs. This procedure is initialized by setting an arbitrary trajectory which is consistent with the evidence. It then alternates between randomly picking a component X_i and sampling a trajectory from the distribution of X_i conditioned on the trajectories of the other components and the evidence. This procedure is reminiscent of *block Gibbs sampling* (Gilks et al., 1996) as we sample an entire trajectory rather than a single random variable in each iteration. However, in our approach we need to sample a continuous trajectory.

The crux of our approach is in the way we sample a trajectory for a single component from a process that is conditioned on trajectories of the other components. While such a process is Markovian, it is not homogeneous as its dynamics depends on trajectories of its Markov Blanket as well as on past and present evidence. We show that we can perform exact sampling by utilizing this Markovian property, and that the cost of this procedure is determined by the complexity of the current trajectories and the sampled one, and not by a pre-defined resolution parameter. This implies that the computational time adapts to the complexity of the sampled object.

2 Continuous-Time Bayesian Networks

In this section we briefly review the CTBN model (Nodelman et al., 2002). Consider an M-component Markov process

$$\boldsymbol{X}^{(t)} = (X_1^{(t)}, X_2^{(t)}, \ldots X_M^{(t)})$$

with state space $S = S_1 \times S_2 \times \cdots \times S_M$.

A notational convention: vectors are denoted by boldface symbols, e.g., $\boldsymbol{X}, \boldsymbol{a}$, and matrices are denoted by blackboard style characters, e.g., $\mathbb{Q}$. The states in S are denoted by vectors of indexes, $\boldsymbol{a} = (a_1, \ldots, a_M)$. The indexes $1 \leq i, j \leq M$ are used to enumerate the components. We use the notation $\boldsymbol{X}^{(t)}$ and $X_i^{(t)}$ to denote a random variable at time t. We will use $\boldsymbol{X}^{[s,t]}$, $\boldsymbol{X}^{(s,t]}$, $\boldsymbol{X}^{[s,t)}$, to denote the state of $\boldsymbol{X}$ in the closed and semi-open intervals from s to t.

The dynamics of a time-homogeneous continuous-time Markov process are fully determined by the *Markov transition function*,

$$p_{\boldsymbol{a},\boldsymbol{b}}(t) = \Pr(\boldsymbol{X}^{(t+s)} = \boldsymbol{b} | \boldsymbol{X}^{(s)} = \boldsymbol{a}),$$

where time-homogeneity implies that the right-hand side does not depend on s. Provided that the transition function satisfies certain analytical properties (continuity, and regularity; see Chung (1960)) the dynamics are fully captured by a constant matrix $\mathbb{Q}$—the *rate*, or *intensity matrix*—whose entries $q_{\boldsymbol{a},\boldsymbol{b}}$ are defined by

$$q_{\boldsymbol{a},\boldsymbol{b}} = \lim_{h \downarrow 0} \frac{p_{\boldsymbol{a},\boldsymbol{b}}(h) - \delta_{\boldsymbol{a},\boldsymbol{b}}}{h},$$

where $\delta_{\boldsymbol{a},\boldsymbol{b}}$ is a multivariate Kronecker delta.

A Markov process can also be viewed as a generative process: The process starts in some state $\boldsymbol{a}$. After spending a finite amount of time at $\boldsymbol{a}$, it transitions, at a random time, to a random state $\boldsymbol{b} \neq \boldsymbol{a}$. The transition times to the various states are exponentially distributed, with rate parameters $q_{\boldsymbol{a},\boldsymbol{b}}$. The diagonal elements of $\mathbb{Q}$ are set such that each row sums up to zero.

The time-dependent probability distribution, $\boldsymbol{p}(t)$, whose entries are defined by

$$p_{\boldsymbol{a}}(t) = \Pr(\boldsymbol{X}^{(t)} = \boldsymbol{a}), \qquad \boldsymbol{a} \in S,$$

satisfies the so-called *forward*, or *master*, *equation*,

$$\frac{d\boldsymbol{p}}{dt} = \mathbb{Q}^T \boldsymbol{p}. \tag{1}$$

Thus, using the $\mathbb{Q}$ matrix, we can write the Markov transition function as

$$p_{\boldsymbol{a},\boldsymbol{b}}(t) = [\exp(t\mathbb{Q})]_{\boldsymbol{a},\boldsymbol{b}},$$

that is, as the $\boldsymbol{a}, \boldsymbol{b}$ entry in the matrix resulting from exponentiating $\mathbb{Q}$ (using matrix exponentiation).

It is important to note that the master Eq. (1) encompasses all the statistical properties of the Markov process. There is a one-to-one correspondence between the description of a Markov process by means of a master equation, and by means of a "pathwise" characterization (up to stochastic equivalence of the latter; see Gikhman and Skorokhod (1975)).

Continuous-time Bayesian Networks provide a compact representation of multi-component Markov processes by incorporating two assumptions: (1) every transition involves a single component; (2) each component undergoes transitions at a rate which depends only on the state of a subsystem of components.

Formally, the structure of a CTBN is defined by assigning to each component i a set of indices $\mathrm{Par}(i) \subseteq$

$\{1,\ldots,M\}\setminus\{i\}$. With each component i, we associate a *conditional rate matrix* $\mathbb{Q}^{i|\,\mathrm{Par}(i)}$ with entries $q^{i|\,\mathrm{Par}(i)}_{a_i,b_i|\boldsymbol{u}_i}$ where a_i and b_i are states of X_i and $\boldsymbol{u}_i$ is a state of $\mathrm{Par}(i)$. This matrix defines the rate of X_i as a function of the state of its parents. Thus, when the parents of X_i change state, the rates governing its transition can change.

The formal semantics of CTBNs is in terms of a joint rate matrix for the whole process. This rate matrix is defined by combining the conditional rate matrices

$$q_{\boldsymbol{a},\boldsymbol{b}} = \sum_{i=1}^{M}\left(q^{i|\,\mathrm{Par}(i)}_{a_i,b_i|\,\mathrm{P}_i(\boldsymbol{a})}\prod_{j\neq i}\delta_{a_j,b_j}\right). \qquad (2)$$

where $\mathrm{P}_i(\boldsymbol{a})$ is a projection operator that project a complete assignment $\boldsymbol{a}$ to an assignment to the $\mathrm{Par}(i)$ components. Eq. (2) is, using the terminology of Nodelman et al. (2002), the "amalgamation" of the M conditional rate matrices. Note the compact representation, which is valid for both diagonal and off-diagonal entries. It is also noteworthy that amalgamation is a summation, rather than a product; indeed, independent exponential rates are additive. If, for example, every component has d possible values and k parents, the rate matrix requires only $Md^{k+1}(d-1)$ parameters, rather than $d^M(d^M-1)$.

The dependency relations between components can be represented graphically as a directed graph, $\mathcal{G}$, in which each node corresponds to a component, and each directed edge defines a parent-child relation. A CTBN consists of such a graph, supplemented with a set of M conditional rate matrices $\mathbb{Q}^{i|\,\mathrm{Par}(i)}$. The graph structure has two main roles: (i) it provides a data structure to which parameters are associated; (ii) it provides a qualitative description of dependencies among the various components of the system. The graph structure also reveals conditional independencies between sets of components (Nodelman et al., 2002).

Notational conventions: Full trajectories and observed pointwise values of components are denoted by lower case letters indexed by the relevant time intervals, e.g., $x_i^{(t)}$, $x_i^{[s,t]}$. We will use $\Pr(x_i^{(t)})$ and $\Pr(x_i^{[s,t]})$ as shorthands for $\Pr(X_i^{(t)}=x_i^{(t)})$ and $\Pr(X_i^{[s,t]}=x_i^{[s,t]})$.

It should be emphasized that even though CTBNs provide a succinct representation of multi-component processes, any inference query still requires the exponentiation of the full $d^M\times d^M$ dimensional rate matrix $\mathbb{Q}$. For example, given the state of the system at times 0 and T, the *Markov bridge* formula is

$$\Pr(\boldsymbol{X}^{(t)}=\boldsymbol{a}|\boldsymbol{x}^{(0)},\boldsymbol{x}^{(T)}) = \frac{[\exp(t\mathbb{Q})]_{\boldsymbol{x}^{(0)},\boldsymbol{a}}[\exp((T-t)\mathbb{Q})]_{\boldsymbol{a},\boldsymbol{x}^{(T)}}}{[\exp(T\mathbb{Q})]_{\boldsymbol{x}^{(0)},\boldsymbol{x}^{(T)}}}.$$

It is the premise of this work that such expressions cannot be computed directly, thus requiring approximation algorithms.

3 Sampling in a Two Component Process

3.1 Introduction

We will start by addressing the task of sampling from a two components process. The generalization to multi-component processes will follow in the next section.

Consider a two-component CTBN, $\boldsymbol{X}=(X,Y)$, whose dynamics is defined by conditional rates $\mathbb{Q}^{X|Y}$ and $\mathbb{Q}^{Y|X}$ (that is, X is a parent of Y and Y is a parent of X). Suppose that we are given partial evidence about the state of the system. This evidence might contain point observations, as well as continuous observations in some intervals, of the states of one or two components. Our goal is to sample a trajectory of (X,Y) from the joint posterior distribution.

The approach we take here is to use a Gibbs sampler (Gilks et al., 1996) over trajectories. In such a sampler, we initialize X and Y with trajectories that are consistent with the evidence. Then, we randomly either sample a trajectory of X given the entire trajectory of Y and the evidence on X, or sample a trajectory of Y given the entire trajectory of X and the evidence on Y. This procedure defines a random walk in the space of (X,Y) trajectories. The basic theory of Gibbs sampling suggests that this random walk will converge to the distribution of X,Y given the evidence.

To implement such a sampler, we need to be able to sample the trajectory of one component given the entire trajectory of the other component and the evidence. Suppose, we have a fully observed trajectory on Y. In this case, observations on X at the extremities of some time interval statistically separate this interval from the rest of trajectory. Thus, we can restrict our analysis to the following situation: the process is restricted to a time interval $[0,T]$ and we are given observations $X^{(0)}=x^{(0)}$ and $X^{(T)}=x^{(T)}$, along with the entire trajectory of Y in $[0,T]$. The latter consists of a sequence of states $(y_0,\ldots,y_K)$ and transition times $(\tau_0=0,\tau_1,\ldots,\tau_K,\tau_{K+1}=T)$. An example of such scenario is shown in Figure 1(a). The entire problem is now reduced to the following question: how can we sample a trajectory of X in the interval $(0,T)$ from its posterior distribution?

To approach this problem we exploit the fact that *the sub-process X given that $Y^{[0,T]}=y^{[0,T]}$ is Markovian* (although non-homogeneous in time):

Proposition 3.1: *The following Markov property holds for all $t>s$,*

$$\Pr(X^{(t)}\mid x^{[0,s]},x^{(T)},y^{[0,T]}) = \Pr(X^{(t)}\mid x^{(s)},x^{(T)},y^{[s,T]}).$$

3.2 Time Granularized Process

Analysis of such process requires reasoning about a continuum of random variables. A natural way of doing so is to perform the analysis in discrete time with a finite time granularity h, and examine the behavior of the system when we take $h\downarrow 0$.

To do so, we introduce some definitions. Suppose $\Pr$ is the probability function associated with a continuous-time Markov process with rate matrix $\mathbb{Q}$. We define the *h-coarsening* of $\Pr$ to be $\Pr_h$, a distribution over the random variables $\boldsymbol{X}^{(0)}, \boldsymbol{X}^{(h)}, \boldsymbol{X}^{(2h)}, \ldots$ which is defined by the dynamics

$$\Pr_h(\boldsymbol{X}^{(t+h)} = \boldsymbol{b} \mid \boldsymbol{X}^{(t)} = \boldsymbol{a}) = \delta_{\boldsymbol{a},\boldsymbol{b}} + h \cdot q_{\boldsymbol{a},\boldsymbol{b}},$$

which is the Taylor expansion of $[\exp(t\mathbb{Q})]_{\boldsymbol{a},\boldsymbol{b}}$, truncated at the linear term. When $h < \min_{\boldsymbol{a}}(-1/q_{\boldsymbol{a},\boldsymbol{a}})$, $\Pr_h$ is a well-defined distribution.

We would like to show that the measure $\Pr_h(A)$ of an event A converges to $\Pr(A)$ when $h \downarrow 0$. To do so, however, we need to define the h-coarsening of an event. Given a time point t, define $\lfloor t\rfloor_h$ and $\lceil t\rceil_h$ to be the rounding down and up of t to the nearest multiple of h. For point events we define $[\![\boldsymbol{X}^{(t)} = \boldsymbol{a}]\!]_h$ to be the event $\boldsymbol{X}^{(\lfloor t\rfloor_h)} = \boldsymbol{a}$, and $[\![\boldsymbol{X}^{(t^+)} = \boldsymbol{a}]\!]_h$ to the event $\boldsymbol{X}^{(\lceil t\rceil_h)} = \boldsymbol{a}$. For an interval event, we define $[\![\boldsymbol{X}^{(s,t]} = \boldsymbol{a}_{(s,t]}]\!]_h$ to be the event $\boldsymbol{X}^{(\lceil s\rceil_h)} = \boldsymbol{a}_{\lceil s\rceil_h}, \boldsymbol{X}^{(\lceil s\rceil_h+h)} = \boldsymbol{a}_{\lceil s\rceil_h+h}, \ldots, \boldsymbol{X}^{(\lfloor t\rfloor_h)} = \boldsymbol{a}_{\lfloor t\rfloor_h}$. Similarly, we can define the coarsening of events over only one component and composite events.

Note that the probability of any given trajectory tends to zero as $h \to 0$. The difficulty in working directly in the continuous-time formulation is that we condition on events that have zero probability. The introduction of a granularized process allows us to manipulate well-defined conditional probabilities, which remain finite as $h \to 0$.

Theorem 3.2: Let A and B be point, interval, or a finite combination of such events. Then

$$\lim_{h\downarrow 0} \Pr_h([\![A]\!]_h \mid [\![B]\!]_h) = \Pr(A \mid B)$$

From now on, we will drop the $[\![A]\!]_h$ notation, and assume it implicitly in the scope of $\Pr_h()$.

A simple minded approach to solve our problem is to work with a given finite h and use discrete sampling to sample trajectories in the coarsened model (thus, working with a *dynamical Bayesian network*). If h is sufficiently small this might be a reasonable approximation to the desired distribution. However, this approach suffers from sub-optimality due to this fixed time granularity — a too coarse granularity leads to inaccuracies, while a too fine granularity leads to computational overhead. Moreover, when different components evolve at different rates, this trade-off is governed by the fastest component.

3.3 Sampling a Continuous-Time Trajectory

To avoid the trade-offs of fixed time granularity we exploit the fact that while a single trajectory is defined over infinite time points, it involves only a finite number of transitions in a finite interval. Therefore, instead of sampling states at different time points, we only sample a finite sequence of transitions. The Markovian property of the conditional process X enables doing so using a sequential procedure.

Our procedure starts by sampling the first transition time. It then samples the new state the transition leads to. As this new sample point statistically separates the remaining interval from the past, we are back with the initial problem yet with a shorter interval. We repeat these steps until the entire trajectory is sampled; it terminates once the next transition time is past the end of the interval.

Our task is to sample the first transition time and the next state, conditioned on $X^{(0)} = x^{(0)}$, $X^{(T)} = x^{(T)}$ as well as the entire trajectory of Y in $[0, T]$. To sample this transition time, we first define the conditional cumulative distribution function $F(t)$ that X stays in the initial state for a time less than t:

$$F(t) = 1 - \Pr\left(X^{(0,t]} = x^{(0)} | x^{(0)}, x^{(T)}, y^{[0,T]}\right) \tag{3}$$

If we can evaluate this function, then we can sample the first transition time τ by inverse transform sampling — we draw ξ from a uniform distribution in the interval $[0, 1]$, and set $\tau = F^{-1}(\xi)$; see Figure 1a,b.

The Markov property of the conditional process allows us to decompose the probability that X remains in its initial state until time t. Denoting the probability of Y's trajectory and of X remaining in its initial state until time t by

$$p^{\text{past}}(t) = \Pr(X^{(0,t]} = x^{(0)}, y^{(0,t]} | x^{(0)}, y^{(0)}),$$

and the probability of future observations given the state of (X_t, Y_t) by

$$p_x^{\text{future}}(t) = \Pr(x^{(T)}, y^{(t,T]} | X^{(t)} = x, y^{(t)}).$$

We can then write the probability that X is in state $x^{(0)}$ until t as

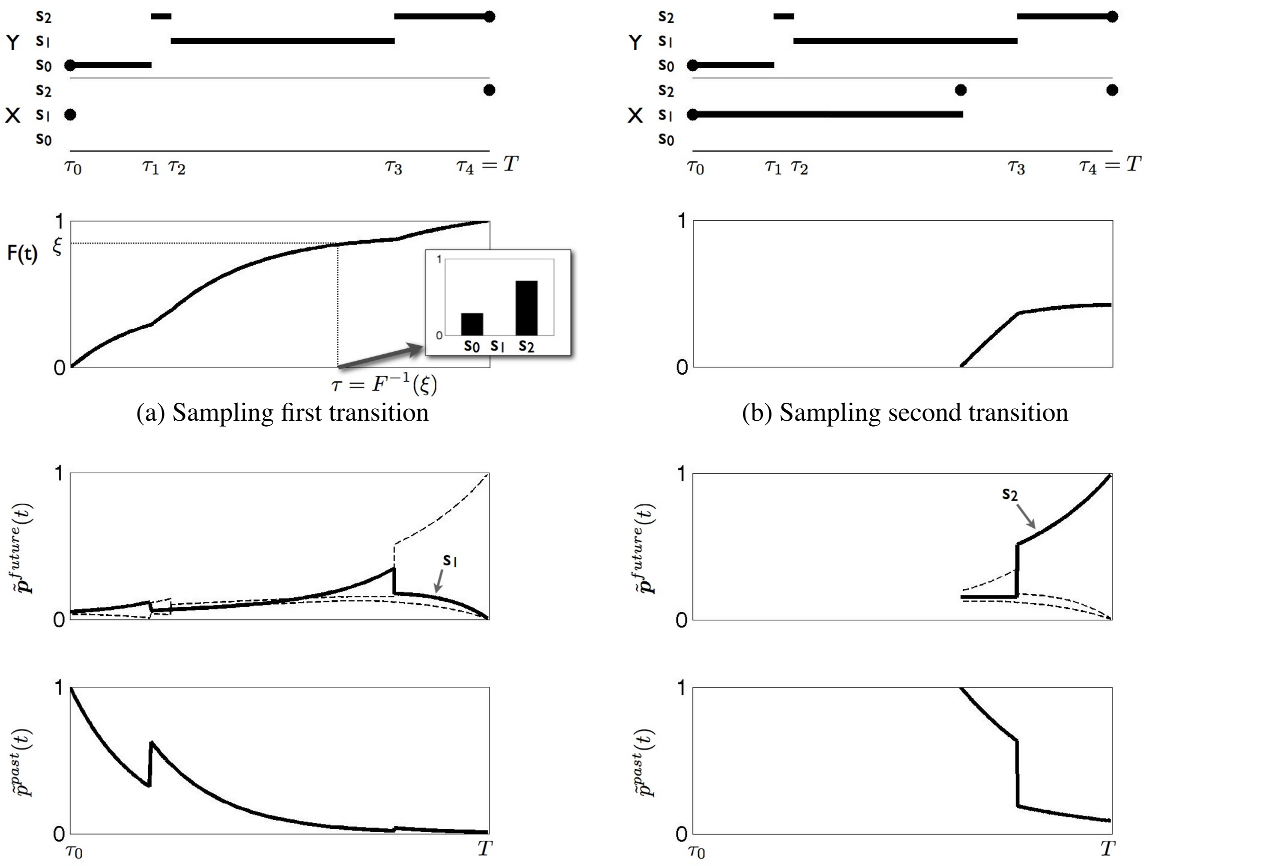
$$\Pr\left(X^{(0,t]} = x^{(0)} | x^{(0)}, x^{(T)}, y^{[0,T]}\right) = \frac{p^{\text{past}}(t) \cdot p_{x^{(0)}}^{\text{future}}(t)}{p_{x^{(0)}}^{\text{future}}(0)}. \tag{4}$$

Lamentably, while the reasoning we just described is seemingly correct, all the terms in Eq. (4) are equal to 0, since they account for the probability of Y's trajectory. However, as we shall see, if we evaluate this equation carefully we will be able to define it with terms that decompose the problem in a similar manner.

To efficiently compute these terms we exploit the fact that although the process is not homogeneous, the dynamics of the joint process within an interval $[\tau_k, \tau_{k+1})$, in which Y has a fixed value y_k, is characterized by a *single* unnormalized rate matrix whose entries depend on y_k. This allows us to adopt a *forward-backward* propagation scheme. We now develop the details of these propagations.

3.4 Computing $p^{\text{past}}(t)$

We begin with expressing $p^{\text{past}}(t)$ as a product of local terms. Recall that $p^{\text{past}}(t)$ is the probability that X is constant until time t. We denote by $p_h^{\text{past}}(t)$ the h-coarsened version of $p^{\text{past}}(t)$.



(a) Sampling first transition

(b) Sampling second transition

(c) Initial propagators

(d) Propagators in second step

Figure 1: Illustration of sampling of a single component with three states. (a) Top panel: sampling scenario, with a complete trajectory for Y, that has four transitions at $\tau_1, \ldots, \tau_4$, and point evidence on X at times 0 and T. Bottom panel: the cumulative distribution $F(t)$, that X changes states before time t given this evidence. We sample the next transition time by drawing ξ from a uniform distribution and setting $\tau = F^{-1}(\xi)$. Note that as $x^{(0)} \neq x^{(T)}$, $F(T) = 1$. The bar graph represents the conditional distribution of the next state, given a transition at time τ. (b) Same sampling procedure for the second transition. Here $F(T) < 1$ since it is not necessary for X to change its state. (c and d) The two components used in computing $1 - F(t)$: $\tilde{p}^{\text{past}}(t)$ the probability that X stays with a constant value until time t and Y has the observed trajectory until this time; and $\tilde{p}_t^{\text{future}}(x)$ the probability that X transition's from state x at t to its observed state at time T and Y follows its trajectory from t to T.

To characterize the dynamics within intervals (τ_k, τ_{k+1}) we define *constant propagator functions*

$$\phi^y_{h,x}(\Delta t) = \\ \Pr_h(X^{(t,t+\Delta t]} = x, Y^{(t,t+\Delta t]} = y | X^{(t)} = x, Y^{(t)} = y)$$

These functions determine the probability that $X = x$ and $Y = y$ throughout an interval of length Δt if they start with these values.

At time τ_{k+1} the Y component changes it value from y_k to y_{k+1}. The transition probability at this point is $h \cdot q^{Y|X}_{y_k,y_{k+1}|x^{(0)}}$. Thus, from the Markov property of the joint process it follows that for $t \in (\tau_k, \tau_{k+1})$

$$p^{\text{past}}_h(t) = \left[\prod_{l=0}^{k-1} \phi^{y_l}_{h,x^{(0)}}(\Delta_l) \cdot q^{Y|X}_{y_l,y_{l+1}|x^{(0)}} \cdot h\right] \phi^{y_k}_{h,x^{(0)}}(t-\tau_k)$$

where $\Delta_l = \tau_{l+1} - \tau_l$.

To compute the constant propagator functions, we realize that in each step within the interval $(s, t]$ the state does not change. Thus,

$$\phi^y_{h,x}(\Delta t) = [1 + h \cdot (q^{X|Y}_{x,x|y} + q^{Y|X}_{y,y|x})]^{\frac{\lfloor \Delta t \rfloor_h}{h}}$$

We define

$$\phi^y_x(\Delta t) = \lim_{h \downarrow 0} \phi^y_{h,x}(\Delta t) = e^{(\Delta t)(q^{X|Y}_{x,x|y} + q^{Y|X}_{y,y|x})}$$

We conclude that if

$$\tilde{p}^{\text{past}}(t) = \left[\prod_{l=0}^{k-1} \phi_{x^{(0)}}^{y_l}(\Delta_l) \cdot q_{y_l,y_{l+1}|x^{(0)}}^{Y|X}\right] \phi_{x^{(0)}}^{y_k}(t-\tau_k),$$

then for $t \in (\tau_k, \tau_{k+1})$

$$\lim_{h \downarrow 0} \frac{p_h^{\text{past}}(t)}{h^k} = \tilde{p}^{\text{past}}(t)$$

3.5 Computing $p_x^{\text{future}}(t)$

We now turn to computing $p_x^{\text{future}}(t)$. Unlike the previous case, here we need to compute this term for every possible value of x. We do so by backward dynamic programing (reminiscent of backward messages in HMMs).

We denote by $\boldsymbol{p}_h^{\text{future}}(t)$ a vector with entries $p_{h,x}^{\text{future}}(t)$. Note that, $\boldsymbol{p}_h^{\text{future}}(T) = \boldsymbol{e}_{x^{(T)}}$ where $\boldsymbol{e}_x$ is the unit vector with 1 in position x. Next, we define a *propagator matrix* $\mathbb{S}_h^y(\Delta t)$ with entries

$$s_{h,a,b}^y(\Delta t) = \\ \Pr_h(X^{(t+\Delta t)} = b, Y^{(t,t+\Delta t]} = y | X^{(t)} = a, Y^{(t)} = y)$$

This matrix provides the dynamics of X in an interval where Y is constant. We can use it to compute the probability of transitions between states of X in the intervals $(\tau_k, \tau_{k+1}]$, for every $\tau_k < s < t < \tau_{k+1}$

$$\boldsymbol{p}_h^{\text{future}}(s) = \mathbb{S}_h^{y_k}(t-s)\boldsymbol{p}_h^{\text{future}}(t)$$

At transition points τ_k we need to take into account the probability of a change. To account for such transitions, we define a diagonal matrix $\mathbb{T}^{y,y'}$ whose (a,a) entry is $q_{y,y'|a}^{Y|X}$. Using this notation and the Markov property of the joint process the conditional probability of future observations for $\tau_k \le t \le \tau_{k+1}$ is

$$\boldsymbol{p}_h^{\text{future}}(t) = \\ \mathbb{S}_h^{y_{k-1}}(\tau_{k+1}-t)\left[\prod_{l=k+1}^{K} h\mathbb{T}^{y_l,y_{l+1}}\mathbb{S}_h^y(\Delta_l)\right]\boldsymbol{e}_{x^{(T)}}$$

It remains to determine the form of the propagator matrix. At time granularity h, we can write the probability of transitions between states of X while $Y = y$ as a product of transition matrices. Thus,

$$\mathbb{S}_h^y(\Delta t) = (I + h \cdot \mathbb{R}_{X|y})^{\frac{\lfloor \Delta t \rfloor_h}{h}}$$

where $\mathbb{R}^{X|y}$ is the matrix with entries

$$r_{a,b}^{X|y} = \begin{cases} q_{a,b|y}^{X|Y} & a \neq b \\ \\ q_{a,a|y}^{X|Y} + q_{y,y|a}^{Y|X} & a = b \end{cases}$$

We now can define

$$\mathbb{S}^y(\Delta t) = \lim_{h \downarrow 0} \mathbb{S}_h^y(\Delta t) = e^{(\Delta t)\mathbb{R}^{X|y}}$$

This terms is similar to transition matrix of a Markov process. Note, however that $\mathbb{R}$ is not a stochastic rate matrix, as the rows do not sum up to 0. In fact, the sum of the rows in negative, which implies that the entries in $\mathbb{S}_h^y(\Delta t)$ tend to get smaller with Δt. This matches the intuition that this term should capture the probability of the evidence that $Y = y$ for the whole interval.

To summarize, if we define for $t \in (\tau_k, \tau_{k+1})$

$$\tilde{\boldsymbol{p}}^{\text{future}}(t) = \mathbb{S}^{y_{k-1}}(\tau_{k+1}-t)\left[\prod_{l=k+1}^{K} \mathbb{T}^{y_l,y_{l+1}}\mathbb{S}^y(\Delta_l)\right]\boldsymbol{e}_{x^{(T)}},$$

then

$$\lim_{h \downarrow 0} \frac{\boldsymbol{p}_h^{\text{future}}(t)}{h^{K-k}} = \tilde{\boldsymbol{p}}^{\text{future}}(t)$$

3.6 Putting it All Together

Based on the above arguments.

$$\Pr_h\left(X^{(0,t]} = x^{(0)} | x^{(0)}, x^{(T)}, y^{[0,T]}\right) = \frac{p_h^{\text{past}}(t) p_{h,x^{(0)}}^{\text{future}}(t)}{p_{h,x^{(0)}}^{\text{future}}(0)}$$

Now, if $t \in (\tau_k, \tau_{k+1})$, then

$$\begin{aligned} &\Pr\left(X^{(0,t]} = x^{(0)} | x^{(0)}, x^{(T)}, y^{[0,T]}\right) \\ &= \lim_{h \downarrow 0} \frac{p_h^{\text{past}}(t) p_{h,x^{(0)}}^{\text{future}}(t)}{p_{h,x^{(0)}}^{\text{future}}(0)} \\ &= \lim_{h \downarrow 0} \frac{[h^{-k} p_h^{\text{past}}(t)][h^{-(K-k)} p_{h,x^{(0)}}^{\text{future}}(t)]}{h^{-K} p_{h,x^{(0)}}^{\text{future}}(0)} \\ &= \frac{\tilde{p}^{\text{past}}(t)\tilde{p}_{x^{(0)}}^{\text{future}}(t)}{\tilde{p}_{x^{(0)}}^{\text{future}}(0)} \end{aligned}$$

Thus, in both numerator and denominator we must account for the observation of K transitions of Y, which have probability of $o(h^K)$. Since these term cancels out, we remain with the conditional probability over the event of interest.

3.7 Forward Sampling

To sample an entire trajectory we first compute $\tilde{\boldsymbol{p}}^{\text{future}}(t)$ only at transition times from the final transition to the start.

We sample the first transition time by drawing a random value ξ from a uniform distribution in $[0,1]$. Now we find τ such that $F(\tau) = \xi$ in two steps: First, we sequentially search for the interval $[\tau_k, \tau_{k+1}]$ such that $F(\tau_k) \le F(\tau) \le F(\tau_{k+1})$ by propagating $\tilde{p}^{\text{past}}(t)$ forward through transition points. Second, we search the exact time point within $[\tau_k, \tau_{k+1}]$ using binary search with L steps to obtain accuracy of $2^{-L}\Delta_k$. This step requires computation of $\mathbb{S}^{y_k}(2^{-L}\Delta_k)$ and its exponents $\mathbb{S}^{y_k}(2^{-l}\Delta_k)$, $l = 1, \ldots, L-1$.

Once we sample the transition time t, we need to compute the probability of the new state of X. Using similar arguments as the ones we discussed above, we find that

$$\Pr\left(X^{(t^+)} = x | X^{[0,t)} = x^{(0)}, X^{(t^+)} \neq x^{(0)}, y^{[0,T]}\right) = \frac{q^{X|Y}_{x^{(0)},x} \cdot \tilde{p}^{\text{future}}_x(t)}{\sum_{x' \neq x^{(0)}} q^{X|Y}_{x^{(0)},x'} \cdot \tilde{p}^{\text{future}}_{x'}(t)}.$$

Thus, we can sample the next state by using the precomputed value of $\tilde{p}^{\text{future}}_x(t)$ at t.

Once we sample a transition (time and state), we can sample the next transition in the interval $[\tau, T]$. The procedure proceeds while exploiting propagators which have already been computed. It stops when $F(T) < \xi$, i.e., the next sampled transition time is greater than T. Figure 1 illustrates the conditional distributions of the first two transitions.

4 Sampling in a Multi-Component Process

The generalization from a two-component process to a general one is relatively straightforward. At each step, we need to sample a single component X_i conditioned on trajectories in $\boldsymbol{Y} = (X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_M)$. To save computations we exploit the fact that given complete trajectories over the Markov blanket of X_i, which is the component set of X_i's parents, children and its children's parents, the dynamics in X_i is independent of the dynamics of all other components (Nodelman et al., 2002).

Indeed, the structured representation of a CTBN allows computations using only terms involving the Markov blanket. To see that, we first notice that within an interval whose state is $\boldsymbol{Y}_t = \boldsymbol{y}$ the propagator matrix involves terms which depend only on the parents of X_i $q^{X_i|Y}_{a,b|\boldsymbol{y}} = q^{X_i|\text{Par}(i)}_{a,b|\boldsymbol{u}_i}$ and terms which depend on the other members of the Markov blanket,

$$q^{\boldsymbol{Y}|X_i}_{\boldsymbol{y},\boldsymbol{y}|x_i} = \sum_{j \in \text{Child}(i)} q^{X_j|\text{Par}(j)}_{x_j,x_j|\boldsymbol{u}_j} + c_{\boldsymbol{y}}$$

where $c_{\boldsymbol{y}}$ does not depend on the state of X_i. Therefore, we define the reduced rate matrix $\mathbb{R}_{X_i|\boldsymbol{v}}$:

$$r^{X_i|\text{MB}(i)}_{a,b|\boldsymbol{v}} = \begin{cases} q^{X_i|\text{Par}(i)}_{a,b|\boldsymbol{u}_i} & a \neq b \\ q^{X_i|\text{Par}(i)}_{a,a|\boldsymbol{u}_i} + \sum_{j \in \text{Child}(i)} q^{X_j|\text{Par}(j)}_{x_j,x_j|\boldsymbol{u}_j} & a = b \end{cases}$$

where, $\boldsymbol{v}$ is the projection of $\boldsymbol{y}$ to the Markov blanket. Consequently the local propagator matrix becomes

$$\mathbb{S}^{\boldsymbol{v}}(t) = \exp(t \cdot \mathbb{R}_{X_i|\boldsymbol{v}}) \tag{5}$$

Importantly, this matrix differs from $\mathbb{S}^{\boldsymbol{y}}(t)$ by a scalar factor of $\exp(t \cdot c_{\boldsymbol{y}})$. The same factor arise when replacing the term in the exponent of the constant propagator. Therefore, these terms cancel out upon normalization.

This development also shows that when sampling X_i we only care about transition points of one of the trajectories in $\text{MB}(i)$. Thus, the intervals computed in the initial backward propagation are defined by these transitions. Therefore, the complexity of the backward procedure scales with the rate of X_i and its Markov blanket.

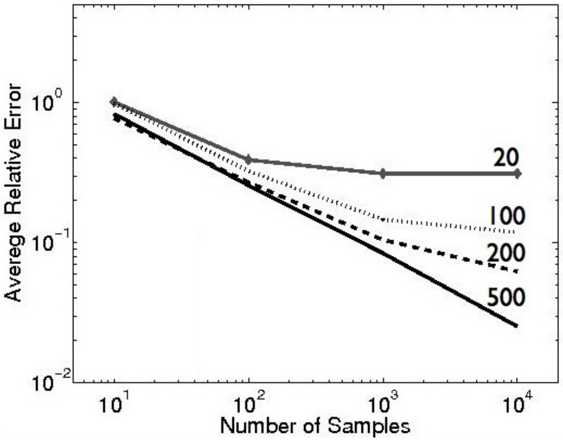


Figure 2: Relative error versus burn-in and number of samples.

5 Experimental Evaluation

We evaluate convergence properties of our procedure on a chain network presented in Fan and Shelton (2008), as well as on related networks of various sizes and parametrizations. The basic network contains 5 components, $X_0, \rightarrow X_1 \rightarrow \ldots X_4$, with 5 states each. The transition rates of X_0 suggest a tendency to cycle in 2 possible loops: $s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow s_0$ and $s_0 \rightarrow s_3 \rightarrow s_4 \rightarrow s_0$; whereas for $i > 0$, X_i attempts to follow the state of X_{i-1} — the transition $q^{X_i|X_{i-1}}_{a,b|c}$ has higher intensity when $c = b$. The intensities of X_0 in the original network are symmetric relative to the two loops. We slightly perturbed parameters to break symmetry since the symmetry between the two loops tends to yield untypically fast convergence.

To obtain a reliable convergence assessment, we should generate samples from multiple independent chains which are initialized from an over-dispersed distribution. Aiming to construct such samples, our initialization procedure draws for each component a rate matrix by choosing an assignment to its parents from a uniform distribution and taking the corresponding conditional rate matrix. Using these matrices it samples a trajectory that is consistent with evidence independently for every component using the backward propagation-forward sampling strategy we described above.

A crucial issue in MCMC sampling is the time it takes the chain to *mix* — that is, sample from a distribution that is close to the target distribution rather than the initial distribution. It is not easy to show empirically that a chain has mixed. We examine this issue from a pragmatic perspective by asking what is the quality of the estimates based on sam-

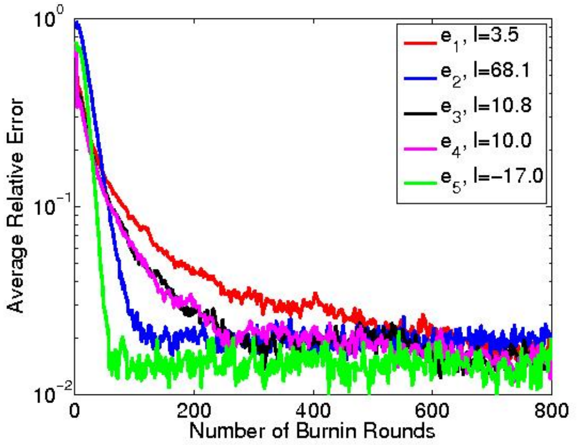


Figure 3: Error versus burn-in for different evidence sets. For each set we specify the average log-likelihood of the samples after convergence.

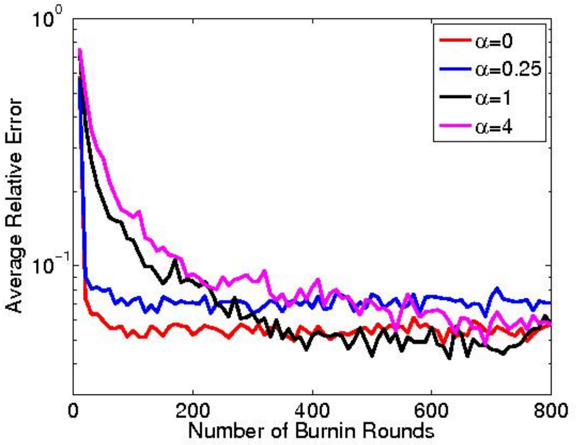


Figure 4: Effect of conditional transition probability sharpness on mixing time.

ples taken at different number of "burn-in" iterations after the initialization, where a single iteration involves sampling each of the components once. We examine the estimates of expected sufficient statistics that are required for learning CTBN's — residence time of components in states and the number of transitions given the state of the component's parent (Nodelman et al., 2003). We measure estimation quality by the *average relative error* $\sum_j \frac{|\hat{\theta}_j - \theta_j|}{\theta_j}$ where θ_j is exact value of the j'th sufficient statistics calculated using numerical integration and $\hat{\theta}_j$ is the approximation.

To make the task harder, we chose an extreme case by setting evidence $\boldsymbol{X}^{(0)} = \vec{s}_0$ (the vector of s_0), and $\boldsymbol{X}^{(3)} = (s_0, s_1, s_3, s_0, s_1)$. We then sampled the process using multiple random starting points, computed estimated expected statistics, and compared them the exact expected statistics. Figure 2 shows the behavior of the average relative error taken over all $\theta > 0.05$ versus the sample size for different number of burn-in iterations. Note that when using longer burn-in, the error decreases at a rate of $O(\sqrt{n})$, where n is the number of samples, which is what we would expect from theory, if the samples where totally independent. This implies that at this long burn-in the error due to the sampling process is smaller than the error contributed by the number of samples.

To study further the effect of evidence's likelihood, we measured error versus burn-in using 10,000 samples in our original evidence set, and four additional ones. The first additional evidence, denoted by $\boldsymbol{e}_2$ is generated by setting $\boldsymbol{X}^{(0)} = \vec{s}_0$, forward sampling a random trajectory and taking the complete trajectory of X_4 as evidence. Additional sets are: $\boldsymbol{e}_3 = \{\boldsymbol{X}^{(0)} = \vec{s}_0, \boldsymbol{X}^{(3)} = \vec{s}_0\}$; $\boldsymbol{e}_4 = \{\boldsymbol{X}^{(0)} = \vec{s}_0\}$ and an extremely unlikely case $\boldsymbol{e}_5 = \{\boldsymbol{X}^{(0)} = \vec{s}_0, X_0^{(0,3)} = s_0, \boldsymbol{X}^{(3)} = (s_0, s_1, s_3, s_0, s_1)\}$.

Figure 3 illustrates that burn-in period may vary by an order of magnitude, however it is not correlated with the log-likelihood. Note that in this specific experiment slower convergence occurs when continuous evidence is absent. The reason for this may be the existence of multiple possible paths that cycle through state zero. That is, the posterior distribution is , in a sense, multi-modal.

To further explore the effect of the posterior's landscape, we tested networks with similar total rate of transitions, but with varying level of coupling between components. Stronger coupling of components leads to a sharper joint distribution. To achieve variations in the coupling we consider variants of the chain CTBN where we set $\hat{\pi}_{a,b|y} = \frac{(q_{a,b|y})^\alpha}{\sum_{c \neq a}(q_{a,c|y})^\alpha}$ and $\hat{q}_{a,b|y} = q_{a,a|y} \cdot \hat{\pi}_{a,b|y}$ where α is a non-negative sharpness parameter As $\alpha \to 0$ the network becomes smoother, which reduces coupling between components. However, the stationary distribution is not tending to a uniform one because we do not alter the diagonal elements. Figure 4 shows convergence behavior for different values of α where estimated statistics are averaged over 1,000 samplers. As we might expect, convergence is faster as the network becomes smoother.

Next we evaluated the scalability of the algorithm by generating networks containing additional components with an architecture similar to the basic chain network. As exact inference is infeasible in such networks we measured relative error versus estimations taken from long runs. Specifically, for each N, we generated 1000 samples by running 100 independent chains and taking samples after 10,000 rounds as well as additional 9 samples from each chain every 1,000 rounds. Using these samples we estimated the target sufficient statistics. To avoid averaging different numbers of components, we compared the relative error in the estimate of 5 components for networks of different sizes. Figure 5 shows the results of this experiment. As we can see, convergence rates decay moderately

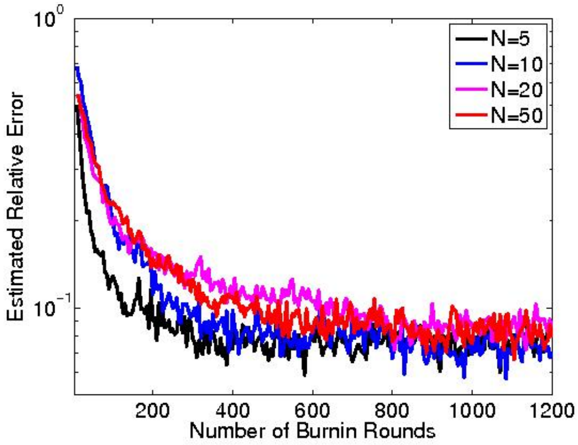


Figure 5: Convergence of relative error in statistics of first five components in networks of various sizes. Errors are computed with respect to statistics that are generated with $N = 10,000$ rounds.

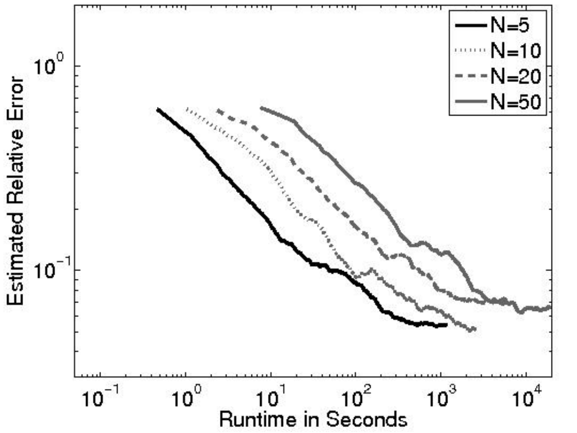


Figure 6: Relative error versus run-time in seconds for various network sizes.

with the size of the network.

While for experimental purposes we generate many samples independently. A practical strategy is to run a small number of chains in parallel and then collect take a large number of samples from each. We tested this strategy by generating 10 independent chain for various networks and estimating statistics from all samples except the first 20%. Using these, we measured how the behavior of error versus CPU run-time scales with network size. Average results of 9 independent tests are shown in Figure 6. Roughly, the run-time required for a certain level of accuracy scales linearly with network size.

Our sampling procedure is such that the cost of sampling a component depends on the time scales of its Markov neighbors and its own rate matrix. To demonstrate that, we created a chain network where each component has rates that are of half the magnitude of its parent. This means that the first component tends to switch state twice as fast as the second, the second is twice as fast as the third, and so on. When we examine the number of transitions in the sampled trajectories Figure 7, we see that indeed they are consistent with these rates, and quickly converge to the expected number, since in this example the evidence is relatively weak. When we examine the number of intervals in the Markov blanket of each components, again we see that neighbors of fast components have more intervals. In this graph X_1 is an anomaly since it does not have a parent.

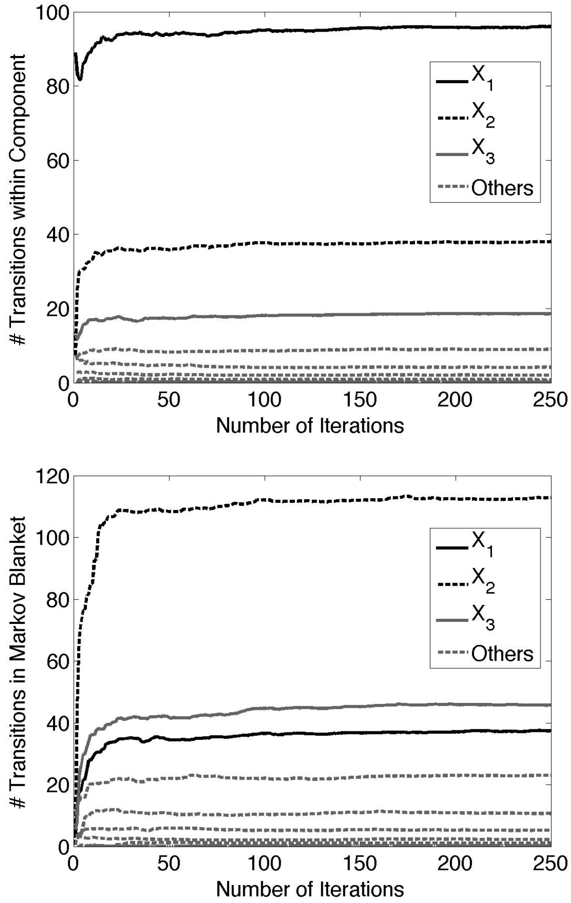


Figure 7: The effect of different time scales on the sampling. In this network X_i's rate is twice as fast than X_{i+1}'s rate. (top) The number transitions sampled for each of the first four components as a function of iteration number. (bottom) The number of intervals of Markov neighbors of each component as a function of iteration number.

6 Discussion

In this paper we presented a new approach for approximate inference in Continuous-Time Bayesian Networks. By building on the strategy of Gibbs sampling. The core of our method is a new procedure for exact sampling of a trajectory of a single component, given evidence on its end points and the full trajectories of its Markov blanket components. This sampling procedure adapts in a natural way to the time scale of the component, and is exact, up to a predefined resolution, without sacrificing efficiency.

This is the first MCMC sampling procedure for this type of models. As such it provides an approach that can sample from the exact posterior, even for unlikely evidence. As the current portfolio of inference procedures for continuous-time processes is very small, our procedure provides another important tool for addressing these models. In particular, since the approach is *asymptotically unbiased* in the number of iterations it can be used to judge the systematic bias introduced by other, potentially faster, approximate inference methodologies, such as the one of Saria et al. (2007).

It is clear that sampling complete trajectories is not useful in situations where we expect a very large number of transitions in the relevant time periods. However, in many applications of interest, and in particular our long term goal of modeling sequence evolution (El-Hay et al., 2006), this is not the case. When one or few components transitions much faster than neighboring components, then we are essentially interested in its average behavior (Friedman and Kupferman, 2006). In such situations, it would be useful to develop a Rao-Blackwellized sampler that integrates over the fast components.

As with many MCMC procedures, one of the main concerns is the mixing time of the sampler. An important direction for future research is the examination of methods for accelerating the mixing - such as *Metropolis-coupled MCMC* or *simulated tempering* (Gilks et al., 1996) - as well as a better theoretic understanding of the convergence properties.

Acknowledgments

We thank Ido Cohn and the anonymous reviewers for helpful remarks on previous versions of the manuscript. This research was supported in part by grants from the Israel Science Foundation and the US-Israel Binational Science Foundation. Tal El-Hay is supported by the Eshkol fellowship from the Israeli Ministry of Science.

References

Chung, K. (1960). *Markov chains with stationary transition probabilities*. Springer Verlag, Berlin.

El-Hay, T., Friedman, N., Koller, D., and Kupferman, R. (2006). Continuous time markov networks. In *Proceedings of the Twenty-second Conference on Uncertainty in AI (UAI)*.

Fan, Y. and Shelton, C. (2008). Sampling for approximate inference in continuous time Bayesian networks. In *Tenth International Symposium on Artificial Intelligence and Mathematics*.

Friedman, N. and Kupferman, R. (2006). Dimension reduction in singularly perturbed continuous-time Bayesian networks. In *Proceedings of the Twenty-second Conference on Uncertainty in AI (UAI)*.

Gardiner, C. (2004). *Handbook of stochastic methods*. Springer-Verlag, New-York, third edition.

Gikhman, I. and Skorokhod, A. (1975). *The theory of Stochastic processes II*. Springer Verlag, Berlin.

Gilks, W. R., S., R., and J., S. D. (1996). *Markov Chain Monte Carlo in Practice*. Chapman & Hall.

Heskes, T. and Zoeter, O. (2002). Expectation propagation for approximate inference in dynamic Bayesian networks. In *Uncertainty in Artificial Intelligence: Proceedings of the Eighteenth Conference (UAI-2002)*, pages 216–233.

Minka, T. P. (2001). Expectation propagation for approximate Bayesian inference. In *Proc. Seventeenth Conference on Uncertainty in Artificial Intelligence (UAI '01)*, pages 362–369.

Ng, B., Pfeffer, A., and Dearden, R. (2005). Continuous time particle filtering. In *Proceedings of the 19th International Joint Conference on AI*.

Nodelman, U., Shelton, C., and Koller, D. (2002). Continuous time Bayesian networks. In *Proc. Eighteenth Conference on Uncertainty in Artificial Intelligence (UAI '02)*, pages 378–387.

Nodelman, U., Shelton, C., and Koller, D. (2003). Learning continuous time Bayesian networks. In *Proc. Nineteenth Conference on Uncertainty in Artificial Intelligence (UAI '03)*, pages 451–458.

Nodelman, U., Shelton, C., and Koller, D. (2005). Expectation propagation for continuous time Bayesian networks. In *Proc. Twenty-first Conference on Uncertainty in Artificial Intelligence (UAI '05)*, pages 431–440.

Saria, S., Nodelman, U., and Koller, D. (2007). Reasoning at the right time granularity. In *Proceedings of the Twenty-third Conference on Uncertainty in AI (UAI)*.

Chapter 4

Paper: Mean field variational approximation for continuous-time Bayesian networks

Ido Cohn[*], Tal El-Hay[*] , Nir Friedman, and Raz Kupferman

Journal of Machine Learning Research, 11(Oct):27452783, 2010

[*] These authors contributed equally.

Journal of Machine Learning Research 11 (2010) 2745-2783 Submitted 5/10; Revised 9/10; Published 10/10

Mean Field Variational Approximation for Continuous-Time Bayesian Networks[*]

Ido Cohn[†] IDO_COHN@CS.HUJI.AC.IL
Tal El-Hay[†] TALE@CS.HUJI.AC.IL
Nir Friedman NIR@CS.HUJI.AC.IL
School of Computer Science and Engineering
The Hebrew University
Jerusalem 91904, Israel

Raz Kupferman RAZ@MATH.HUJI.AC.IL
Institute of Mathematics
The Hebrew University
Jerusalem 91904, Israel

Editor: Manfred Opper

Abstract

Continuous-time Bayesian networks is a natural structured representation language for multi-component stochastic processes that evolve continuously over time. Despite the compact representation provided by this language, inference in such models is intractable even in relatively simple structured networks. We introduce a mean field variational approximation in which we use a product of *inhomogeneous* Markov processes to approximate a joint distribution over trajectories. This variational approach leads to a globally consistent distribution, which can be efficiently queried. Additionally, it provides a lower bound on the probability of observations, thus making it attractive for learning tasks. Here we describe the theoretical foundations for the approximation, an efficient implementation that exploits the wide range of highly optimized ordinary differential equations (ODE) solvers, experimentally explore characterizations of processes for which this approximation is suitable, and show applications to a large-scale real-world inference problem.

Keywords: continuous time Markov processes, continuous time Bayesian networks, variational approximations, mean field approximation

1. Introduction

Many real-life processes can be naturally thought of as evolving continuously in time. Examples cover a diverse range, starting with classical and modern physics, but also including robotics (Ng et al., 2005), computer networks (Simma et al., 2008), social networks (Fan and Shelton, 2009), gene expression (Lipshtat et al., 2005), biological evolution (El-Hay et al., 2006), and ecological systems (Opper and Sanguinetti, 2007). A joint characteristic of all above examples is that they are complex systems composed of multiple components (e.g., many servers in a server farm and multiple residues in a protein sequence). To realistically model such processes and use them in

[*] A preliminary version of this paper appeared in the Proceedings of the Twenty Fifth Conference on Uncertainty in Artificial Intelligence, 2009 (UAI 09).
[†] These authors contributed equally.

making sensible predictions we need to learn how to reason about systems that are composed of multiple components and evolve continuously in time.

Generally, when an evolving system is modeled with sufficient detail, its evolution in time is Markovian; meaning that its future state it determined by its present state—whether in a deterministic or random sense—independently of its past states. A traditional approach to modeling a multi-component Markovian process is to discretize the entire time interval into regular time slices of fixed length and represent its evolution using a *Dynamic Bayesian network*, which compactly represents probabilistic transitions between consecutive time slices (Dean and Kanazawa, 1989; Murphy, 2002; Koller and Friedman, 2009). However, as thoroughly explained in Nodelman et al. (2003), discretization of a time interval often leads either to modeling inaccuracies or to an unnecessary computational overhead. Therefore, in recent years there is a growing interest in modeling and reasoning about multi-component stochastic processes in continuous time (Nodelman et al., 2002; Ng et al., 2005; Rajaram et al., 2005; Gopalratnam et al., 2005; Opper and Sanguinetti, 2007; Archambeau et al., 2007; Simma et al., 2008).

In this paper we focus on *continuous-time Markov processes* having a discrete product state space $S = S_1 \times S_2 \times \cdots \times S_D$, where D is the number of components and the size of each S_i is finite. The dynamics of such processes that are also *time-homogeneous* can be determined by a single rate matrix whose entries encode transition rates among states. However, as the size of the state space is exponential in the number of components so does the size of the transition matrix. *Continuous-time Bayesian networks* (CTBNs) provide an elegant and compact representation language for multi-component processes that have a sparse pattern of interactions (Nodelman et al., 2002). Such patterns are encoded in CTBNs using a directed graph whose nodes represent components and edges represent direct influences among them. The instantaneous dynamics of each component depends only on the state of its parents in the graph, allowing a representation whose size scales linearly with the number of components and exponentially only with the indegree of the nodes of the graph.

Inference in multi-component temporal models is a notoriously hard problem (Koller and Friedman, 2009). Similar to the situation in discrete time processes, inference in CTBNs is exponential in the number of components, even with sparse interactions (Nodelman et al., 2002). Thus, we have to resort to approximate inference methods. The recent literature has adapted several strategies from discrete graphical models to CTBNs in a manner that attempts to exploit the continuous-time representation, thereby avoiding the drawbacks of discretizing the model.

One class of approximations includes sampling-based approaches, where Fan and Shelton (2008) introduce a likelihood-weighted sampling scheme, and more recently El-Hay et al. (2008) introduce a Gibbs-sampling procedure. The complexity of the Gibbs sampling procedure has been shown to naturally adapt to the rate of each individual component. Additionally it yields more accurate answers with the investment of additional computation. However, it is hard to bound the required time in advance, tune the stopping criteria, or estimate the error of the approximation.

An alternative class of approximations is based on *variational principles*. Recently, Nodelman et al. (2005b) and Saria et al. (2007) introduced an *Expectation Propagation* approach, which can be roughly described as a local message passing scheme, where each message describes the dynamics of a single component over an interval. This message passing procedure can be efficient. Moreover it can automatically refine the number of intervals according to the complexity of the underlying system. Nonetheless, it does suffer from several caveats. On the formal level, the approximation has no convergence guarantees. Second, upon convergence, the computed marginals do not necessarily form a globally consistent distribution. Third, it is restricted to approximations in the form of piecewise-homogeneous messages on each interval. Thus, the refinement of the number of intervals depends on the fit of such homogeneous approximations to the target process. Finally, the approximation of Nodelman *et al* does not provide a provable approximation on the likelihood of the observation—a crucial component in learning procedures.

Here, we develop an alternative variational approximation, which provides a different tradeoff. We use the strategy of structured variational approximations in graphical models (Jordan et al., 1999), and specifically the variational approach of Opper and Sanguinetti (2007) for approximate inference in latent Markov Jump Processes, a related class of models (see below for more elaborate comparison). The resulting procedure approximates the posterior distribution of the CTBN as a product of independent components, each of which is an inhomogeneous continuous-time Markov process. We introduce a novel representation that is both natural and allows numerically stable computations. By using this representation, we derive an iterative variational procedure that employs passing information between neighboring components as well as solving a small set of differential equations (ODEs) in each iteration. The latter allows us to employ highly optimized standard ODE solvers in the implementation. Such solvers use an adaptive step size, which as we show is more efficient than any fixed time interval approximation.

We finally describe how to extend the proposed procedure to branching processes and particularly to models of molecular evolution, which describe historical dynamics of biological sequences that employ many interacting components. Our experiments on this domain demonstrate that our procedure provides a good approximation both for the likelihood of the evidence and for the expected sufficient statistics. In particular, the approximation provides a lower-bound on the likelihood, and thus is attractive for use in learning.

The paper is organized as follows: In Section 2 we review continuous-time models and inference problems in such models. Section 3 introduces a general variational principle for inference using a novel parameterization. In Section 4 we apply this principle to a family of factored representations and show how to find an optimal approximation within this family. Section 5 discusses related work. Section 6 gives an initial evaluation. Section 7 presents branching process and further experiments, and Section 8 discusses our results.

2. Foundations

CTBNs are based on the framework of *continuous-time Markov processes (CTMPs)*. In this section we begin by briefly describing CTMPs. See, for example, Gardiner (2004) and Chung (1960) for a thorough introduction. Next we review the semantics of CTBNs. We then discuss inference problems in CTBNs and the challenges they pose.

2.1 Continuous Time Markov Processes

A *continuous-time stochastic process with state space S* is an uncountable collection of S-valued random variables $\{X^{(t)} : t \geq 0\}$ where $X^{(t)}$ describes the state of the system at time t. Systems with multiple components are described by state spaces that are Cartesian products of spaces, S_i, each representing the state of a single component. In this paper we consider a D-component stochastic process $X^{(t)} = (X_1^{(t)}, \ldots, X_D^{(t)})$ with state space $S = S_1 \times S_2 \times \ldots \times S_D$, where each S_i is finite. The states in S are denoted by vectors, $x = (x_1, \ldots, x_D)$.

A *continuous-time Markov process* is a continuous-time stochastic process in which the joint distribution of every finite subset of random variables $X^{(t_0)}, X^{(t_1)}, \dots, X^{(t_K)}$, where $t_0 < t_1 < \cdots < t_K$, satisfies the conditional independence property, also known as the Markov property:

$$\Pr(X^{(t_K)} = x_K | X^{(t_{K-1})} = x_{K-1}, \dots, X^{(t_0)} = x_0) = \Pr(X^{(t_K)} = x_K | X^{(t_{K-1})} = x_{K-1}).$$

In simple terms, the knowledge of the state of the system at a certain time make its states at later times independent of its states at former times. In that case the distribution of the process is fully determined by the conditional probabilities of random variable pairs $\Pr(X^{(t+s)} = y | X^{(s)} = x)$, namely, by the probability that the process is in state y at time $t+s$ given that is was in state x at time s, for all $0 \leq s < t$ and $x, y \in S$. A CTMP is called *time homogeneous* if these conditional probabilities do not depend on s but only on the length of the time interval t, thus, the distribution of the process is determined by the *Markov transition functions*,

$$p_{x,y}(t) \equiv \Pr(X^{(t+s)} = y | X^{(s)} = x), \qquad \text{for all } x, y \in S \text{ and } t \geq 0,$$

which for every fixed t can be viewed as the entries of a stochastic matrix indexed by states x and y.

Under mild assumptions on the Markov transition functions $p_{x,y}(t)$, these functions are differentiable. Their derivatives at $t = 0$,

$$q_{x,y} = \lim_{t \to 0^+} \frac{p_{x,y}(t) - \mathbf{1}_{x=y}}{t},$$

are the entries of the *rate matrix* $\mathbb{Q}$, where $\mathbf{1}$ is the indicator function. This rate matrix describes the infinitesimal transition probabilities,

$$p_{x,y}(h) = \mathbf{1}_{x=y} + q_{x,y} h + o(h), \tag{1}$$

where $o(\cdot)$ means decay to zero faster than its argument, that is $\lim_{h \downarrow 0} \frac{o(h)}{h} = 0$. Note that the off-diagonal entries of $\mathbb{Q}$ are non-negative, whereas each of its rows sums up to zero, namely,

$$q_{x,x} = -\sum_{y \neq x} q_{x,y}.$$

The derivative of the Markov transition function for t other than 0 satisfies the so-called *forward*, or *master equation*,

$$\frac{d}{dt} p_{x,y}(t) = \sum_z q_{z,y} p_{x,z}(t). \tag{2}$$

A similar characterization for the time-dependent probability distribution, $p(t)$, whose entries are defined by

$$p_x(t) = \Pr(X^{(t)} = x), \qquad x \in S,$$

is obtained by multiplying the Markov transition function by entries of the initial distribution $p(0)$ and marginalizing, resulting in

$$\frac{d}{dt} p = p \mathbb{Q}. \tag{3}$$

The solution of this ODE is

$$p(t) = p(0) \exp(t \mathbb{Q}),$$

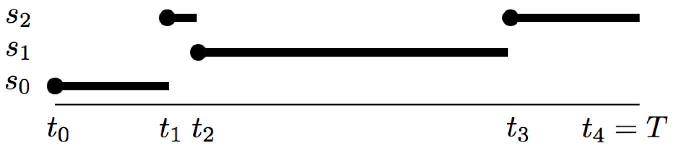


Figure 1: An example of a CTMP trajectory: The process starts at state $x_1 = s_0$, transitions to $x_2 = s_2$ at t_1, to $x_3 = s_1$ at t_2, and finally to $x_4 = s_2$ at t_3.

where $\exp(t\mathbb{Q})$ is a matrix exponential, defined for any square matrix $\mathbb{A}$ by the Taylor series,

$$\exp(\mathbb{A}) = \mathbf{I} + \sum_{k=1}^{\infty} \frac{\mathbb{A}^k}{k!} \ .$$

Applying this solution to the initial condition $p_{x'}(0) = \mathbf{1}_{x=x'}$, we can express the Markov transition function $p_{x,y}(t)$ using the rate matrix $\mathbb{Q}$ as

$$p_{x,y}(t) = [\exp(t\mathbb{Q})]_{x,y}. \tag{4}$$

Although a CTMP is an uncountable collection of random variables (the state of the system at every time t), a *trajectory* σ of $\{X^{(t)}\}_{t\geq 0}$ over a time interval $[0,T]$ can be characterized by a finite number of transitions K, a sequence of states $(x_0, x_1, \ldots, x_K)$ and a sequence of transition times $(t_0 = 0, t_1, \ldots, t_K, t_{K+1} = T)$. We denote by $\sigma(t)$ the state at time t, that is, $\sigma(t) = x_k$ for $t_k \leq t < t_{k+1}$. Figure 1 illustrates such a trajectory.

2.2 Multi-component Representation - Continuous-Time Bayesian Networks

Equation (4) indicates that the distribution of a homogeneous Markov process is fully determined by an initial distribution and a single rate matrix $\mathbb{Q}$. However, since the number of states in a D-component Markov Process is exponential in D, an explicit representation of this transition matrix is often infeasible. *Continuous-time Bayesian networks* are a compact representation of Markov processes that satisfy two assumptions. First it is assumed that only one component can change at a time, thus transition rates involving simultaneous changes of two or more components are zero. Second, the transition rate of each component i depends only on the state of some subset of components denoted $\mathbf{Pa}_i \subseteq \{1,\ldots,D\} \setminus \{i\}$ and on its own state. This dependency is represented using a directed graph, where the nodes are indexed by $\{1,\ldots,D\}$ and the parent nodes of i are $\mathbf{Pa}_i$ (Nodelman et al., 2002). With each component i we then associate a conditional rate matrix $\mathbb{Q}^{i|\mathbf{Pa}_i}_{\cdot|u_i}$ for each state u_i of $\mathbf{Pa}_i$. The off-diagonal entries $q^{i|\mathbf{Pa}_i}_{x_i,y_i|u_i}$ represent the rate at which X_i transitions from state x_i to state y_i given that its parents are in state u_i. The diagonal entries are $q^{i|\mathbf{Pa}_i}_{x_i,x_i|u_i} = -\sum_{y_i \neq x_i} q^{i|\mathbf{Pa}_i}_{x_i,y_i|u_i}$, ensuring that each row in each conditional rate matrix sums up to zero. The dynamics of $X^{(t)}$ are defined by a rate matrix $\mathbb{Q}$ with entries $q_{x,y}$, which combines the conditional rate matrices as follows:

$$q_{x,y} = \begin{cases} q^{i|\mathbf{Pa}_i}_{x_i,y_i|u_i} & \delta(x,y) = \{i\} \\ \sum_i q^{i|\mathbf{Pa}_i}_{x_i,x_i|u_i} & x = y \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

where $\delta(x,y) = \{j|x_j \neq y_j\}$ denotes the set of components in which x differs from y.

To have another perspective on CTBN's, we may consider a discrete-time approximation of the process. Let h be a sampling interval. The subset of random variables $\{X_{t_k} : k \geq 0\}$, where $t_k = kh$, is a discrete-time Markov process over a D-dimensional state-space. *Dynamic Bayesian networks (DBNs)* provide a compact modeling language for such processes, namely the conditional distribution of a DBN $\mathrm{P}_h(X^{(t_k+1)}|X^{(t_k)})$ is factorized into a product of conditional distributions of $X_i^{(t_{k+1})}$ given the state of a subset of $X^{(t_k)} \cup X^{(t_{k+1})}$. When h is sufficiently small, the CTBN can be approximated by a DBN whose parameters depend on the rate matrix $\mathbb{Q}$ of the CTBN ,

$$\mathrm{P}_h(X^{(t_{k+1})} = y|X^{(t_k)} = x) = \prod_{i=1}^{D} \mathrm{P}_h(X_i^{(t_{k+1})} = y_i|X_i^{(t_k)} = x_i, U^{(t_k)} = u_i), \tag{6}$$

where

$$\mathrm{P}_h(X_i^{(t_{k+1})} = y_i|X_i^{(t_k)} = x_i, U^{(t_k)} = u_i) = \mathbf{1}_{x_i=y_i} + q^{i|\mathbf{Pa}_i}_{x_i,y_i|u_i} h. \tag{7}$$

Each such term is the local conditional probability that $X_i^{(t_{k+1})} = y_i$ given the state of X_i and U_i at time t_k. These are valid conditional distributions, because they are non-negative and are normalized, that is

$$\sum_{y_i \in S_i} \left(\mathbf{1}_{x_i=y_i} + q^{i|\mathbf{Pa}_i}_{x_i,y_i|u_i} h \right) = 1$$

for every x_i and u_i. Note that in this discretized process, transition probabilities involving changes in more than one component are $o(h)$, as in the CTBN. Moreover, using Equations (1) and (5) we observe that

$$\Pr(X^{(t_k+1)} = y|X^{(t_k)} = x) = \mathrm{P}_h(X^{(t_k+1)} = y|X^{(t_k)} = x) + o(h).$$

(See Appendix A for detailed derivations). Therefore, the CTBN and the approximating DBN are asymptotically equivalent as $h \to 0$.

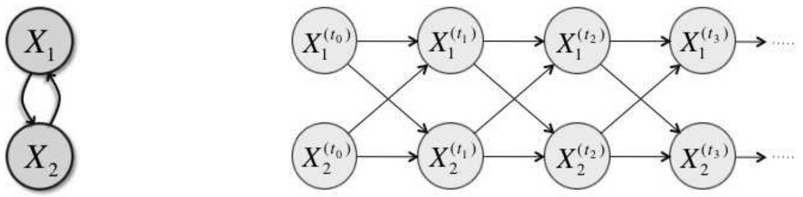
Example 1 An example of a multi-component process is the *dynamic Ising model*, which corresponds to a CTBN in which every component can be in one of two states, -1 or $+1$, and each component prefers to be in the same state as its neighbor. These models are governed by two parameters: a *coupling parameter* β (it is the inverse temperature in physical models, which determines the strength of the coupling between two neighboring components), and a *rate parameter* τ, which determines the propensity of each component to change its state. Low values of β correspond to weak coupling (high temperature). More formally, we define the conditional rate matrices as

$$q^{i|\mathbf{Pa}_i}_{x_i,y_i|u_i} = \tau \left(1 + e^{-2y_i\beta \sum_{j \in \mathbf{Pa}_i} x_j} \right)^{-1}$$

where $x_j \in \{-1,1\}$. This model is derived by plugging the Ising grid to *Continuous-Time Markov Networks*, which are the undirected counterparts of CTBNs (El-Hay et al., 2006).

Consider a two component Ising model whose structure and corresponding DBN are shown in Figure 2. This system is symmetric, that is, the conditional rate matrices are identical for $i \in \{1,2\}$. As an example, for a specific choice of β and τ we have:

$$\mathbb{Q}^{i|\mathbf{Pa}_i}_{\cdot|-1} = \begin{array}{c|cc} & \text{-} & \text{+} \\ \hline \text{-} & -1 & 1 \\ \text{+} & 10 & -10 \end{array} \qquad \mathbb{Q}^{i|\mathbf{Pa}_i}_{\cdot|+1} = \begin{array}{c|cc} & \text{-} & \text{+} \\ \hline \text{-} & -10 & 10 \\ \text{+} & 1 & -1 \end{array}$$



(a) (b)

Figure 2: Two representations of a two binary component dynamic process. *(a)* The associated CTBN. *(b)* The DBN corresponding to the CTBN in (a). The models are equivalent when $h \to 0$.

The local conditional distributions of the DBN can be directly inferred from Equation (7). For example

$$\mathrm{P}_h(X_1^{(t_{k+1})} = 1 | X_1^{(t_k)} = -1, X_2^{(t_k)} = 1) = 10h.$$

Here, in both components the conditional rates are higher for transitions into states that are identical to the state of their parent. Therefore, the two components have a disposition of being in the same state. To support this intuition, we examine the amalgamated rate matrix:

$$\mathbb{Q} \ = \quad \begin{array}{c|cccc} & \texttt{--} & \texttt{-+} & \texttt{+-} & \texttt{++} \\ \hline \texttt{--} & -2 & 1 & 1 & 0 \\ \texttt{-+} & 10 & -20 & 0 & 10 \\ \texttt{+-} & 10 & 0 & -20 & 10 \\ \texttt{++} & 0 & 1 & 1 & -2. \end{array}$$

Clearly, transition rates into states in which both components have the same value is higher. Higher transitions rate imply higher transition probabilities, for example:

$$\begin{array}{rcl} p_{\texttt{-+},\texttt{--}}(h) & = & 10h + o(h), \\ p_{\texttt{--},\texttt{-+}}(h) & = & h + o(h). \end{array}$$

Thus the probability of transitions into a coherent state is much higher than into an incoherent state. ∎

2.3 Inference in Continuous-time Markov Processes

Our setting is as follows: we receive evidence of the states of several or all components within a time interval $[0,T]$. The two possible types of evidence that may be given are continuous evidence, where we know the state of a subset $U \subseteq X$ continuously over some sub-interval $[t_1,t_2] \subseteq [0,T]$, and point evidence of the state of U at some point $t \in [0,T]$. For convenience we restrict our treatment to a time interval $[0,T]$ with full end-point evidence $X^{(0)} = e_0$ and $X^{(T)} = e_T$. We shall discuss the more general case in Section 5.

Given a continuous-time Bayesian network and evidence of the above type we would like to evaluate the likelihood of the evidence, $\Pr(e_0, e_T; \mathbb{Q})$ and to compute pointwise posterior probabilities of various events (e.g., $\Pr(U^{(t)} = u | e_0, e_T)$ for some $U \subseteq X$). Another class of queries are conditional expectations of statistics that involve entire trajectories of the process. Two important examples for queries are the *sufficient statistics* required for learning. These statistics are the amount of time in which X_i is in state x_i and $\mathbf{Pa}_i$ are in state u_i, and the number of transitions that X_i underwent from x_i to y_i while its parents were in state u_i (Nodelman et al., 2003). We denote these statistics by $T^i_{x_i|u_i}$ and $M^i_{x_i,y_i|u_i}$ respectively. For example, in the trajectory of the univariate process in Figure 1, we have $T_{s_2} = t_2 - t_1 + t_4 - t_3$ and $M_{s_0,s_2} = 1$.

Exact calculation of these values is usually a computationally intractable task. For instance, calculation of marginals requires first calculating the pointwise distribution over X using a forward-backward like calculation:

$$\Pr(X^{(t)} = x|e_0, e_T) = \frac{p_{e_0,x}(t)\, p_{x,e_T}(T-t)}{p_{e_0,e_T}(T)}, \tag{8}$$

and then marginalizing

$$\Pr(U^{(t)} = u|e_0, e_T) = \sum_{x\setminus u} \Pr(X^{(t)} = x|e_0, e_T),$$

where $p_{x,y}(t) = [\exp(t\mathbb{Q})]_{x,y}$, and the size of $\mathbb{Q}$ is exponential in the number of components. Moreover, calculating expected residence times and expected number of transitions involves integration over the time interval of these quantities (Nodelman et al., 2005a):

$$\mathbf{E}\left[T_x\right] = \frac{1}{p_{e_0,e_T}(T)} \int_0^T p_{e_0,x}(t)\, p_{x,e_T}(T-t)dt,$$
$$\mathbf{E}\left[M_{x,y}\right] = \frac{1}{p_{e_0,e_T}(T)} \int_0^T p_{e_0,x}(t)\, q_{x,y}\, p_{y,e_T}(T-t)dt \quad .$$

These make this approach infeasible beyond a modest number of components, hence we have to resort to approximations.

3. Variational Principle for Continuous-Time Markov Processes

Variational approximations to structured models aim to approximate a complex distribution by a simpler one, which allows efficient inference. This problem can be viewed as an optimization problem: given a specific model and evidence, find the "best" approximation within a given class of simpler distributions. In this setting the inference is posed as a constrained optimization problem, where the constraints ensure that the parameters correspond to valid distributions consistent with the evidence. Specifically, the optimization problem is constructed by defining a lower bound to the log-likelihood of the evidence, where the gap between the bound and the true likelihood is the divergence of between the approximation and the true posterior. While the resulting problem is generally intractable, it enables us to derive approximate algorithms by approximating either the functional or the constrains that define the set of valid distributions. In this section we define the lower-bound functional in terms of a general continuous-time Markov process (that is, without assuming any network structure). Here we aim at defining a lower bound on $\ln P_{\mathbb{Q}}(e_T|e_0)$ as well as to approximating the posterior probability $P_{\mathbb{Q}}(\cdot \mid e_0, e_T)$, where $P_{\mathbb{Q}}$ is the distribution of the Markov process whose instantaneous rate-matrix is $\mathbb{Q}$. We start by examining the structure of the posterior and introducing an appropriate parameterization.

Recall that the distribution of a time-homogeneous Markov process is characterized by the conditional transition probabilities $p_{x,y}(t)$, which in turn is fully redetermined by the constant rate matrix $\mathbb{Q}$. It is not hard to see that whenever the prior distribution of a stochastic process is that of a homogeneous Markov process with rate matrix $\mathbb{Q}$, then the posterior $P_{\mathbb{Q}}(\cdot|e_0,e_T)$ is also a Markov process, albeit generally not a homogeneous one. The distribution of a continuous-time Markov processes that is not homogeneous in time is determined by conditional transition probabilities, $p_{x,y}(s,s+t)$, which depend explicitly on both initial and final times. These transition probabilities can be expressed by means of a time-dependent matrix-valued function, $\mathbb{R}(t)$, which describes instantaneous transition rates. The connection between the time-dependent rate matrix $\mathbb{R}(t)$ and the transition probabilities, $p_{x,y}(s,s+t)$ is established by the master equation,

$$\frac{d}{dt}p_{x,y}(s,s+t) = \sum_z r_{z,y}(s+t)p_{x,z}(s,s+t),$$

where $r_{z,y}(t)$ are the entries of $\mathbb{R}(t)$. This equation is a generalization of Equation (2) for inhomogeneous processes. As in the homogeneous case, it leads to a master equation for the time-dependent probability distribution,

$$\frac{d}{dt}p_x(t) = \sum_y r_{y,x}(t)p_y(t),$$

thereby generalizing Equation (3).

By the above discussion, it follows that the posterior process can be represented by a time-dependent rate matrix $\mathbb{R}(t)$. More precisely, writing the posterior transition probability using basic properties of conditional probabilities and the definition of the Markov transition function gives

$$P_{\mathbb{Q}}(X^{(t+h)} = y|X^{(t)} = x, X^{(T)} = e_T) = \frac{p_{x,y}(h)p_{y,e_T}(T-t+h)}{p_{x,e_T}(T-t)}.$$

Taking the limit $h \to 0$ we obtain the instantaneous transition rate of the posterior process

$$r_{x,y}(t) = \lim_{h\to 0}\frac{P_{\mathbb{Q}}(X^{(t+h)} = y|X^{(t)} = x, X^{(T)} = e_T)}{h} = q_{x,y}\cdot\frac{p_{y,e_T}(T-t)}{p_{x,e_T}(T-t)}. \tag{9}$$

This representation, although natural, proves problematic in the framework of deterministic evidence because as t approaches T the transition rate into the observed state tends to infinity. In particular, when $x \neq e_T$ and $y = e_T$, the posterior transition rate is $q_{x,e_T}\cdot\frac{p_{e_T,e_T}(T-t)}{p_{x,e_T}(T-t)}$. This term diverges as $t \to T$, because the numerator approaches 1 while the denominator approaches 0. We therefore consider an alternative parameterization for this inhomogeneous process that is more suitable for variational approximations.

3.1 Marginal Density Representation

Let Pr be the distribution of a Markov process, generally not time homogeneous. We define a family of functions:

$$\begin{aligned}\mu_x(t) &= \Pr(X^{(t)} = x),\\ \gamma_{x,y}(t) &= \lim_{h\downarrow 0}\frac{\Pr(X^{(t)} = x, X^{(t+h)} = y)}{h},\ x \neq y.\end{aligned} \tag{10}$$

The function $\mu_x(t)$ is the marginal probability that $X^{(t)} = x$. The function $\gamma_{x,y}(t)$ is the probability density that X transitions from state x to y at time t. Note that this parameter is not a transition rate, but rather a product of a point-wise probability with the point-wise transition rate of the distribution, that is, the entries of the time-dependent rate matrix of an equivalent process can be defined by

$$r_{x,y}(t) = \begin{cases} \frac{\gamma_{x,y}(t)}{\mu_x(t)} & \mu_x(t) > 0, \\ 0 & \mu_x(t) = 0. \end{cases} \tag{11}$$

Hence, unlike the (inhomogeneous) rate matrix at time t, $\gamma_{x,y}(t)$ takes into account the probability of being in state x and not only the rate of transitions.

We aim to use the family of functions μ and γ as a representation of the posterior process. To do so, we need to characterize the set of constraints that these functions satisfy. We begin by constraining the marginals $\mu_x(t)$ to be valid distributions that is, $0 \leq \mu_x(t) \leq 1$ and $\sum_x \mu_x(t) = 1$. A similar constraint on the pairwise distributions implies that $\gamma_{x,y}(t) \geq 0$ for $x \neq y$. Next, we infer additional constraints from consistency properties between distributions over pairs of variables and their uni-variate marginals. Specifically, Equation (10) implies that for $x \neq y$

$$\Pr(X^{(t)} = x, X^{(t+h)} = y) = \gamma_{x,y}(t)\,h + o(h). \tag{12}$$

Plugging this identity into the consistency constraint

$$\mu_x(t) = \Pr(X^{(t)} = x) = \sum_y \Pr(X^{(t)} = x, X^{(t+h)} = y),$$

defining

$$\gamma_{x,x}(t) = -\sum_{y \neq x} \gamma_{x,y}(t)$$

and rearranging, we obtain

$$\Pr(X^{(t)} = x, X^{(t+h)} = y) = \mathbf{1}_{x=y}\mu_x(t) + \gamma_{x,y}(t)\,h + o(h), \tag{13}$$

which unlike (12) is valid for all x, y. Marginalizing (13) with respect to the second variable,

$$\Pr(X^{(t+h)} = x) = \sum_y \Pr(X^{(t)} = y, X^{(t+h)} = x),$$

we obtain a forward update rule for the uni-variate marginals

$$\mu_x(t+h) = \mu_x(t) + h \sum_y \gamma_{y,x}(t) + o(h).$$

Rearranging terms and taking the limit $h \to 0$ gives a differential equation for $\mu_x(t)$,

$$\frac{d}{dt}\mu_x(t) = \sum_y \gamma_{y,x}(t).$$

Finally, whenever $\mu_x(t) = 0$ we have $\Pr(X^{(t)} = x, X^{(t+h)} = y) = 0$, implying in that case that $\gamma_{x,y}(t) = 0$. Based on these observations we define:

Definition 1 A family $\eta = \{\mu_x(t), \gamma_{x,y}(t) : 0 \leq t \leq T\}$ of functions is a *Markov-consistent density set* if the following constraints are fulfilled:

$$\begin{aligned}
\mu_x(t) &\geq 0, \quad \sum_x \mu_x(0) = 1, \\
\gamma_{x,y}(t) &\geq 0 \qquad \forall y \neq x, \\
\gamma_{x,x}(t) &= -\sum_{y \neq x} \gamma_{x,y}(t), \\
\frac{d}{dt}\mu_x(t) &= \sum_y \gamma_{y,x}(t),
\end{aligned}$$

and $\gamma_{x,y}(t) = 0$ whenever $\mu_x(t) = 0$. We denote by $\mathcal{M}$ the set of all Markov-consistent densities. ▮

Using standard arguments we can show that there exists a correspondence between (generally inhomogeneous) Markov processes and density sets η. Specifically, given η, we construct a process by defining an inhomogeneous rate matrix $\mathbb{R}(t)$ whose entries are defined in Equation (11) and prove the following:

Lemma 2 *Let $\eta = \{\mu_x(t), \gamma_{x,y}(t) : 0 \leq t \leq T\}$. If $\eta \in \mathcal{M}$, then there exists a continuous-time Markov process* Pr *for which μ_x and $\gamma_{x,y}$ satisfy (10) for every t in the right-open interval [0,T).*

Proof See appendix B ■

The converse is also true: for every integrable inhomogeneous rate matrix $\mathbb{R}(t)$ the corresponding marginal density set is defined by $\frac{d}{dt}\mu_x(t) = \sum_y r_{y,x}(t)\mu_y(t)$ and $\gamma_{x,y}(t) = \mu_x(t) r_{x,y}(t)$. The processes we are interested in, however, have additional structure, as they correspond to the posterior distribution of a time-homogeneous process with end-point evidence. In that case, multiplying Equation (9) by $\mu_x(t)$ gives

$$\gamma_{x,y}(t) = \mu_x(t) \cdot q_{x,y} \cdot \frac{p_{y,e_T}(T-t)}{p_{x,e_T}(T-t)}. \tag{14}$$

Plugging in Equation (8) we obtain

$$\gamma_{x,y}(t) = \frac{p_{e_0,x}(t) \cdot q_{x,y} \cdot p_{y,e_T}(T-t)}{p_{e_0,e_T(T)}},$$

which is zero when $y \neq e_T$ and $t = T$. This additional structure implies that we should only consider a subset of $\mathcal{M}$. Specifically the representation η corresponding to the posterior distribution $P_{\mathbb{Q}}(\cdot|e_0, e_T)$ satisfies $\mu_x(0) = \mathbf{1}_{x=e_0}$, $\mu_x(T) = \mathbf{1}_{x=e_T}$, $\gamma_{x,y}(0) = 0$ for all $x \neq e_0$ and $\gamma_{x,y}(T) = 0$ for all $y \neq e_T$. We denote by $\mathcal{M}_e \subset \mathcal{M}$ the subset that contains Markov-consistent density sets satisfying these constraints. This analysis suggests that for every homogeneous rate matrix and point evidence e there is a member in $\mathcal{M}_e$ that corresponds to the posterior process. Thus, from now on we restrict our attention to density sets from $\mathcal{M}_e$.

3.2 Variational Principle

The marginal density representation allows us to state the variational principle for continuous processes, which closely tracks similar principles for discrete processes. Specifically, we define a functional of functions that are constrained to be density sets from $\mathcal{M}_e$. The maximum over this set is the log-likelihood of the evidence and is attained for a density set that represents the posterior distribution. This formulation will serve as a basis for the mean-field approximation, which is introduced in the next section.

Define a *free energy functional*,

$$\mathcal{F}(\eta;\mathbb{Q}) = \mathcal{E}(\eta;\mathbb{Q}) + \mathcal{H}(\eta),$$

which, as we will see, measures the quality of η as an approximation of $P_{\mathbb{Q}}(\cdot|e)$. (For succinctness, we will assume that the evidence e is clear from the context.) The two terms in the functional are the *average energy*,

$$\mathcal{E}(\eta;\mathbb{Q}) = \int_0^T \sum_x \left[\mu_x(t) q_{x,x} + \sum_{y\neq x} \gamma_{x,y}(t) \ln q_{x,y} \right] dt,$$

and the *entropy*,

$$\mathcal{H}(\eta) = \int_0^T \sum_x \sum_{y\neq x} \gamma_{x,y}(t)[1+\ln\mu_x(t) - \ln\gamma_{x,y}(t)]dt.$$

The following theorem establishes the relation of this functional to the Kullback-Leibler (KL) divergence and the likelihood of the evidence, and thus allows us to cast the variational inference into an optimization problem.

Theorem 3 *Let $\mathbb{Q}$ be a rate matrix, $e = (e_0, e_T)$ be states of X, and $\eta \in \mathcal{M}_e$. Then*

$$\mathcal{F}(\eta;\mathbb{Q}) = \ln P_{\mathbb{Q}}(e_T|e_0) - \boldsymbol{D}(P_\eta \| P_{\mathbb{Q}}(\cdot|e))$$

where P_η is the distribution corresponding to η and $\boldsymbol{D}(P_\eta\|P_{\mathbb{Q}}(\cdot|e))$ is the KL divergence between the two processes.

We conclude from the non-negativity of the KL divergence that the energy functional $\mathcal{F}(\eta;\mathbb{Q})$ is a lower bound of the log-likelihood of the evidence. The closer the approximation to the target posterior, the tighter the bound. Moreover, since the KL divergence is zero if and only if the two distributions are equal almost everywhere, finding the maximizer of this free energy is equivalent to finding the posterior distribution from which answers to different queries can be efficiently computed.

3.3 Proof of Theorem 3

We begin by examining properties of distributions of inhomogeneous Markov processes. Let $X^{(t)}$ be an inhomogeneous Markov process with rate matrix $\mathbb{R}(t)$. As in the homogeneous case, a trajectory σ of $\{X^{(t)}\}_{t\geq 0}$ over a time interval $[0,T]$ can be characterized by a finite number of transitions K, a sequence of states $(x_0, x_1, \ldots, x_K)$ and a sequence of transition times $(t_0 = 0, t_1, \ldots, t_K, t_{K+1} = T)$. We denote by Σ the set of all trajectories of $X^{[0,T]}$. The distribution over Σ can be characterized by a collection of random variables that consists of the number of transitions κ, a sequence of states $(\chi_0, \ldots, \chi_\kappa)$ and transition times $(\tau_1, \ldots, \tau_\kappa)$. Note that the number of random variables that characterize the trajectory is by itself a random variable. The density $f_{\mathbb{R}}$ of a trajectory $\sigma = \{K, x_0, \ldots, x_K, t_1, \ldots, t_K\}$ is the derivative of the joint distribution with respect to transition times, that is,

$$f_{\mathbb{R}}(\sigma) = \frac{\partial^K}{\partial t_1 \cdots \partial t_K} P_{\mathbb{R}}(\kappa = K, \chi_0 = x_0, \ldots, \chi_K = x_K, \tau_1 \leq t_1, \ldots, \tau_K \leq t_K),$$

which is given by

$$f_{\mathbb{R}}(\sigma) = p_{x_0}(0) \cdot \prod_{k=0}^{K-1} \left[e^{\int_{t_k}^{t_{k+1}} r_{x_k,x_k}(t)dt} r_{x_k,x_{k+1}}(t_{k+1}) \right] \cdot e^{\int_{t_K}^{t_{K+1}} r_{x_K,x_K}(t)dt}.$$

For example, in case $\mathbb{R}(t) = \mathbb{Q}$ is a homogeneous rate matrix this equation reduces to

$$f_{\mathbb{Q}}(\sigma) = p_{x_0}(0) \cdot \prod_{k=0}^{K-1} \left[e^{q_{x_k,x_k}(t_{k+1}-t_k)} q_{x_k,x_{k+1}} \right] \cdot e^{q_{x_K,x_K}(t_{K+1}-t_K)}.$$

The expectation of a random variable $\psi(\sigma)$ is an infinite sum (because one has to account for all possible numbers of transitions) of finite dimensional integrals,

$$\mathbf{E}_{f_{\mathbb{Q}}}[\psi] \equiv \int_{\Sigma} f_{\mathbb{R}}(\sigma)\psi(\sigma)d\sigma \equiv \sum_{K=0}^{\infty} \sum_{x_0} \cdots \sum_{x_K} \int_0^T \int_0^{t_K} \cdots \int_0^{t_2} f_{\mathbb{R}}(\sigma)\psi(\sigma)dt_1 \cdots dt_K.$$

The *KL-divergence* between two densities that correspond to two inhomogeneous Markov processes with rate matrices $\mathbb{R}(t)$ and $\mathbb{S}(t)$ is

$$\boldsymbol{D}(f_{\mathbb{R}} \| f_{\mathbb{S}}) = \int_{\Sigma} f_{\mathbb{R}}(\sigma) \ln \frac{f_{\mathbb{R}}(\sigma)}{f_{\mathbb{S}}(\sigma)} d\sigma \ . \tag{15}$$

We will use the convention $0 \ln 0 = 0$ and assume the support of $f_{\mathbb{S}}$ is contained in the support of $f_{\mathbb{R}}$. That is $f_{\mathbb{R}}(\sigma) = 0$ whenever $f_{\mathbb{S}}(\sigma) = 0$. The KL-divergence satisfies $\boldsymbol{D}(f_{\mathbb{R}} \| f_{\mathbb{S}}) \geq 0$ and is exactly zero if and only if $f_{\mathbb{R}} = f_{\mathbb{S}}$ almost everywhere (Kullback and Leibler, 1951).

Let $\eta \in \mathcal{M}_e$ be a marginal density set consistent with e. As we have seen, this density set corresponds to a Markov process with rate matrix $\mathbb{R}(t)$ whose entries are defined by Equation (11), hence we identify $f_\eta \equiv f_{\mathbb{R}}$.

Given evidence e on some event we denote $f_{\mathbb{Q}}(\sigma, e) \equiv f_{\mathbb{Q}}(\sigma) \cdot \boldsymbol{1}_{\sigma \models e}$, and note that

$$P_{\mathbb{Q}}(e) = \int_{\{\sigma:\sigma\models e\}} f_{\mathbb{Q}}(\sigma)d\sigma = \int_{\Sigma} f_{\mathbb{Q}}(\sigma, e)d\sigma \ ,$$

where $\sigma \models e$ is a predicate which is true if σ is consistent with the evidence. The density function of the posterior distribution $P_{\mathbb{Q}}(\cdot|e)$ satisfies $f_{\mathbb{S}}(\sigma) = \frac{f_{\mathbb{Q}}(\sigma,e)}{P_{\mathbb{Q}}(e)}$ where $\mathbb{S}(t)$ is the time-dependent rate matrix that corresponds to the posterior process.

Manipulating (15), we get

$$\boldsymbol{D}(f_\eta \| f_{\mathbb{S}}) = \int_{\Sigma} f_\eta(\sigma) \ln f_\eta(\sigma) d\sigma - \int_{\Sigma} f_\eta(\sigma) \ln f_{\mathbb{S}}(\sigma) d\sigma \equiv \mathbf{E}_{f_\eta}[\ln f_\eta(\sigma)] - \mathbf{E}_{f_\eta}[\ln f_{\mathbb{S}}(\sigma)].$$

Replacing $\ln f_{\mathbb{S}}(\sigma)$ by $\ln f_{\mathbb{Q}}(\sigma, e) - \ln P_{\mathbb{Q}}(e)$ and applying simple arithmetic operations gives

$$\ln P_{\mathbb{Q}}(e) = \mathbf{E}_{f_\eta}[\ln f_{\mathbb{Q}}(\sigma, e)] - \mathbf{E}_{f_\eta}[\ln f_\eta(\sigma)] + \boldsymbol{D}(f_\eta \| f_{\mathbb{S}}).$$

The crux of the proof is in showing that the expectations in the right-hand side satisfy

$$\mathbf{E}_{f_\eta}[\ln f_{\mathbb{Q}}(\sigma, e)] = \mathcal{E}(\eta; \mathbb{Q}),$$

and

$$-\mathbf{E}_{f_\eta}\left[\ln f_\eta(\sigma)\right] = \mathcal{H}(\eta),$$

implying that $\mathcal{F}(\eta;\mathbb{Q})$ is a lower bound on the log-probability of evidence with equality if and only if $f_\eta = f_\mathbb{Q}$ almost everywhere.

To prove these identities for the energy and entropy, we treat trajectories as functions $\sigma : \mathcal{R} \to \mathcal{R}$ where $\mathcal{R}$ is the set of real numbers by denoting $\sigma(t) \equiv X^{(t)}(\sigma)$—the state of the system at time t. Using this notation we introduce two lemmas that allow us to replace integration over a set of trajectories by a one dimensional integral, which is defined over a time variable. The first result handles expectations of functions that depend on specific states:

Lemma 4 *Let $\psi : S \times \mathcal{R} \to \mathcal{R}$ be a function, then*

$$\mathbf{E}_{f_\eta}\left[\int_0^T \psi(\sigma(t),t)dt\right] = \int_0^T \sum_x \mu_x(t)\psi(x,t)dt.$$

Proof See Appendix C.1 ■

As an example, by setting $\psi(x',t) = \mathbf{1}_{x'=x}$ we obtain that the expected residence time in state x is $\mathbf{E}_{f_\eta}\left[T_x\right] = \int_0^T \mu_x(t)dt$. The second result handles expectations of functions that depend on transitions between states:

Lemma 5 *Let $\psi(x,y,t)$ be a function from $S \times S \times \mathcal{R}$ to $\mathcal{R}$ that is continuous with respect to t and satisfies $\psi(x,x,t) = 0$, $\forall x, \forall t$ then*

$$\mathbf{E}_{f_\eta}\left[\sum_{k=1}^{K^\sigma} \psi(x_{k-1}^\sigma, x_k^\sigma, t_k^\sigma)\right] = \int_0^T \sum_x \sum_{y \neq x} \gamma_{x,y}(t)\psi(x,y,t)dt,$$

where the superscript σ stresses that K^σ, x_k^σ and t_k^σ are associated with a specific trajectory σ.

Proof See Appendix C.2 ■

Continuing the example of the previous lemma, here by setting $\psi(x',y',t) = \mathbf{1}_{x'=x}\mathbf{1}_{y'=y}\mathbf{1}_{x\neq y}$ the sums within the left hand expectation become the number of transitions in a trajectory σ. Thus, we obtain that the expected number of transitions from x to y is $\mathbf{E}_f\left[M_{x,y}\right] = \int_0^T \gamma_{x,y}(t)dt$.

We now use these lemmas to compute the expectations involved in the energy functional. Suppose $e = \{e_0, e_T\}$ is a pair of point evidence and $\eta \in \mathcal{M}_e$. Applying these lemmas with $\psi(x,t) = q_{x,x}$ and $\psi(x,y,t) = \mathbf{1}_{x\neq y} \cdot \ln q_{x,y}$ gives

$$\mathbf{E}_{f_\eta}\left[\ln f_\mathbb{Q}(\sigma,e)\right] = \int_0^T \sum_x \left[\mu_x(t)q_{x,x}(t) + \sum_{y\neq x} \gamma_{x,y}(t)\ln q_{x,y}(t)\right] dt\ .$$

Similarly, setting $\psi(x,t) = r_{x,x}(t)$ and $\psi(x,y,t) = \mathbf{1}_{x\neq y} \cdot \ln r_{x,y}(t)$, where $\mathbb{R}(t)$ is defined in Equation (11), we obtain

$$-\mathbf{E}_{f_\eta}\left[\ln f_\eta(\sigma,e)\right] = -\int_0^T \sum_x \left[\mu_x(t)\frac{\gamma_{x,x}(t)}{\mu_x(t)} + \sum_{y\neq x} \gamma_{x,y}(t)\ln\frac{\gamma_{x,y}(t)}{\mu_x(t)}\right] dt = \mathcal{H}(\eta)\ .$$

4. Factored Approximation

The variational principle we discussed is based on a representation that is as complex as the original process—the number of functions $\gamma_{x,y}(t)$ we consider is equal to the size of the original rate matrix $\mathbb{Q}$. To get a tractable inference procedure we make additional simplifying assumptions on the approximating distribution.

Given a D-component process we consider approximations that factor into products of independent processes. More precisely, we define $\mathcal{M}_e^i$ to be the continuous Markov-consistent density sets over the component X_i, that are consistent with the evidence on X_i at times 0 and T. Given a collection of density sets $\eta^1, \ldots, \eta^D$ for the different components, the product density set $\eta = \eta^1 \times \cdots \times \eta^D$ is defined as

$$\begin{aligned} \mu_x(t) &= \prod_i \mu^i_{x_i}(t), \\ \gamma_{x,y}(t) &= \begin{cases} \gamma^i_{x_i,y_i}(t)\mu_x^{\setminus i}(t) & \delta(x,y) = \{i\} \\ \sum_i \gamma^i_{x_i,x_i}(t)\mu_x^{\setminus i}(t) & x = y \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

where $\mu_x^{\setminus i}(t) = \prod_{j \neq i} \mu^j_{x_j}(t)$ is the joint distribution at time t of all the components other than the i'th (it is not hard to see that if $\eta^i \in \mathcal{M}_e^i$ for all i, then $\eta \in \mathcal{M}_e$). We define the set $\mathcal{M}_e^F$ to contain all factored density sets. From now on we assume that $\eta = \eta^1 \times \cdots \times \eta^D \in \mathcal{M}_e^F$.

Assuming that $\mathbb{Q}$ is defined by a CTBN, and that η is a factored density set, we can rewrite

$$\mathcal{E}(\eta; \mathbb{Q}) = \sum_i \int_0^T \sum_{x_i} \left[\mu^i_{x_i}(t) \mathbf{E}_{\mu^{\setminus i}(t)} \left[q_{x_i,x_i|U_i} \right] + \sum_{x_i, y_i \neq x_i} \gamma^i_{x_i,y_i}(t) \mathbf{E}_{\mu^{\setminus i}(t)} \left[\ln q_{x_i,y_i|U_i} \right] \right] dt$$

(see derivations in Appendix D). Similarly, the entropy term factors as

$$\mathcal{H}(\eta) = \sum_i \mathcal{H}(\eta^i) \ .$$

Note that terms such as $\mathbf{E}_{\mu^{\setminus i}(t)} \left[q_{x_i,x_i|U_i} \right]$ involve only $\mu^j(t)$ for $j \in \mathbf{Pa}_i$, because $\mathbf{E}_{\mu^{\setminus i}(t)} [f(U_i)] = \sum_{u_i} \mu_{u_i}(t) f(u_i)$. Therefore, this decomposition involves only local terms that either include the i'th component, or include the i'th component and its parents in the CTBN defining $\mathbb{Q}$.

To make the factored nature of the approximation explicit in the notation, we write henceforth,

$$\mathcal{F}(\eta; \mathbb{Q}) = \tilde{\mathcal{F}}(\eta^1, \ldots, \eta^D; \mathbb{Q}).$$

4.1 Fixed Point Characterization

The factored form of the functional and the independence between the different η^i allows optimization by *block ascent*, optimizing the functional with respect to each parameter set in turn. To do so, we should solve the following optimization problem:

> Fixing i, and given $\eta^1, \ldots, \eta^{i-1}, \eta^{i+1}, \ldots, \eta^D$, in $\mathcal{M}_e^1, \ldots \mathcal{M}_e^{i-1}, \mathcal{M}_e^{i+1}, \ldots, \mathcal{M}_e^D$, respectively, find
> $$\arg\max_{\eta^i \in \mathcal{M}_e^i} \tilde{\mathcal{F}}(\eta^1, \ldots, \eta^D; \mathbb{Q}) \ .$$

If for all i, we have a $\mu^i \in \mathcal{M}_e^i$, which is a solution to this optimization problem with respect to each component, then we have a (local) stationary point of the energy functional within $\mathcal{M}_e^F$.

To solve this optimization problem, we define a Lagrangian, which includes the constraints in the form of Definition 1. These constraints are to be enforced in a continuous fashion, and so the Lagrange multipliers $\lambda^i_{x_i}(t)$ are continuous functions of t as well. The Lagrangian is a functional of the functions $\mu^i_{x_i}(t)$, $\gamma^i_{x_i,y_i}(t)$ and $\lambda^i_{x_i}(t)$, and takes the following form

$$\mathcal{L} = \tilde{\mathcal{F}}(\eta;\mathbb{Q}) - \sum_{i=1}^{D} \int_0^T \lambda^i_{x_i}(t) \left(\frac{d}{dt}\mu^i_{x_i}(t) - \sum_{y_i} \gamma^i_{x_i,y_i}(t) \right) dt \quad .$$

A necessary condition for the optimality of a density set η is the existence of λ such that (η,λ) is a *stationary point* of the Lagrangian. A stationary point of a functional satisfies the *Euler-Lagrange* equations, namely the *functional derivatives* with respect to μ, γ and λ vanish (see Appendix E for a brief review). The detailed derivation of the resulting equations is in Appendix F. Writing these equations in explicit form, we get a fixed point characterization of the solution in term of the following set of ODEs:

$$\begin{aligned} \frac{d}{dt}\mu^i_{x_i}(t) &= \sum_{y_i \neq x_i} \left(\gamma^i_{y_i,x_i}(t) - \gamma^i_{x_i,y_i}(t) \right), \\ \frac{d}{dt}\rho^i_{x_i}(t) &= -\rho^i_{x_i}(t) \left(\bar{q}^{\,i}_{x_i,x_i}(t) + \psi^i_{x_i}(t) \right) - \sum_{y_i \neq x_i} \rho^i_{y_i}(t) \tilde{q}^i_{x_i,y_i}(t) \end{aligned} \tag{16}$$

along with the following algebraic constraint

$$\rho^i_{x_i}(t)\gamma^i_{x_i,y_i}(t) = \mu^i_{x_i}(t)\tilde{q}^i_{x_i,y_i}(t)\rho^i_{y_i}(t), \; x_i \neq y_i \tag{17}$$

where ρ^i are the exponents of the Lagrange multipliers λ_i. In these equations we use the following shorthand notations for the average rates

$$\begin{aligned} \bar{q}^{\,i}_{x_i,x_i}(t) &= \mathbf{E}_{\mu^{\setminus i}(t)} \left[q^{i|\mathbf{Pa}_i}_{x_i,x_i|U_i} \right], \\ \bar{q}^{\,i}_{x_i,x_i|x_j}(t) &= \mathbf{E}_{\mu^{\setminus i}(t)} \left[q^{i|\mathbf{Pa}_i}_{x_i,x_i|U_i} \mid x_j \right], \end{aligned}$$

where $\mu^{\setminus i}(t)$ is the product distribution of $\mu^1(t),\ldots,\mu^{i-1}(t),\mu^{i+1}(t),\ldots,\mu^D(t)$. Similarly, we have the following shorthand notations for the geometrically-averaged rates,

$$\begin{aligned} \tilde{q}^i_{x_i,y_i}(t) &= \exp\left\{ \mathbf{E}_{\mu^{\setminus i}(t)} \left[\ln q^{i|\mathbf{Pa}_i}_{x_i,y_i|U_i} \right] \right\}, \\ \tilde{q}^i_{x_i,y_i|x_j}(t) &= \exp\left\{ \mathbf{E}_{\mu^{\setminus i}(t)} \left[\ln q^{i|\mathbf{Pa}_i}_{x_i,y_i|U_i} \mid x_j \right] \right\} \quad . \end{aligned}$$

The last auxiliary term is

$$\psi^i_{x_i}(t) = \sum_{j \in Children_i} \sum_{x_j} \left[\mu^j_{x_j}(t)\bar{q}^{\,j}_{x_j,x_j|x_i}(t) + \sum_{x_j \neq y_j} \gamma^j_{x_j,y_j}(t) \ln \tilde{q}^j_{x_j,y_j|x_i}(t) \right] \quad .$$

To uniquely solve the two differential Equations (16) for $\mu^i_{x_i}(t)$ and $\rho^i_{x_i}(t)$ we need to set boundary conditions. The boundary condition for $\mu^i_{x_i}$ is defined explicitly in $\mathcal{M}_e^F$ as

$$\mu^i_{x_i}(0) = \mathbf{1}_{x_i = e_{i,0}} \; . \tag{18}$$

The boundary condition at T is slightly more involved. The constraints in $\mathcal{M}_e^F$ imply that $\mu^i_{x_i}(T) = \mathbf{1}_{x_i=e_{i,T}}$. As stated in Section 3.1, we have that $\gamma^i_{e_{i,T},x_i}(T) = 0$ when $x_i \neq e_{i,T}$. Plugging these values into (17), and assuming that all elements of $\mathbb{Q}^{i|\mathbf{Pa}_i}$ are non-zero we get that $\rho_{x_i}(T) = 0$ for all $x_i \neq e_{i,T}$ (It might be possible to use a weaker condition that $\mathbb{Q}$ is irreducible). In addition, we notice that $\rho_{e_{i,T}}(T) \neq 0$, for otherwise the whole system of equations for ρ will collapse to 0. Finally, notice that the solution of (16,17) for μ^i and γ^i is insensitive to the multiplication of ρ^i by a constant. Thus, we can arbitrarily set $\rho_{e_{i,T}}(T) = 1$, and get the boundary condition

$$\rho^i_{x_i}(T) = \mathbf{1}_{x_i=e_{i,T}}. \tag{19}$$

Putting it all together we obtain a characterization of stationary points of the functional as stated in the following theorem:

Theorem 6 *$\eta^i \in \mathcal{M}_e^i$ is a stationary point (e.g., local maxima) of $\tilde{\mathcal{F}}(\eta^1,\ldots,\eta^D;\mathbb{Q})$ subject to the constraints of Definition 1 if and only if it satisfies (16–19).*

Proof see Appendix F ■

It is straightforward to extend this result to show that at a maximum with respect to all the component densities, this fixed-point characterization must hold for all components simultaneously.

Example 2 Consider the case of a single component, for which our procedure should be exact, as no simplifying assumptions are made on the density set. In that case, the averaged rates $\bar{q}^i$ and the geometrically-averaged rates $\tilde{q}^i$ both reduce to the unaveraged rates q, and $\psi \equiv 0$. Thus, the system of equations to be solved is

$$\frac{d}{dt}\mu_x(t) = \sum_{y\neq x}(\gamma_{y,x}(t) - \gamma_{x,y}(t)),$$

$$\frac{d}{dt}\rho_x(t) = -\sum_y q_{x,y}\rho_y(t),$$

along with the algebraic equation

$$\rho_x(t)\gamma_{x,y}(t) = \mu_x(t)q_{x,y}\rho_y(t), \qquad y \neq x.$$

These equations have a simple intuitive interpretation. First, the backward propagation rule for ρ_x implies that

$$\rho_x(t) = \Pr(e_T|X^{(t)} = x).$$

To prove this identity, we recall the notation $p_{x,y}(h) \equiv \Pr(X^{(t+h)} = y|X^{(t)} = x)$ and write the discretized propagation rule

$$\Pr(e_T|X^{(t)} = x) = \sum_y p_{x,y}(h)\cdot\Pr(e_T|X^{(t+h)} = y)\ .$$

Using the definition of q (Equation 1), rearranging, dividing by h and taking the limit $h \to 0$ gives

$$\frac{d}{dt}\Pr(e_T|X^{(t)} = x) = -\sum_y q_{x,y}\cdot\Pr(e_T|X^{(t)} = y),$$

which is identical to the differential equation for ρ. Second, dividing the above algebraic equation by $\rho_x(t)$ whenever it is greater than zero we obtain

$$\gamma_{x,y}(t) = \mu_x(t) q_{x,y} \frac{\rho_y(t)}{\rho_x(t)}. \tag{20}$$

Thus, we reconstructed Equation (14).

This analysis suggest that this system of ODEs is similar to forward-backward propagation, except that unlike classical forward propagation, here the forward propagation already takes into account the backward messages to directly compute the posterior. Given this interpretation, it is clear that integrating $\rho_x(t)$ from T to 0 followed by integrating $\mu_x(t)$ from 0 to T computes the exact posterior of the processes.

This interpretation of $\rho_x(t)$ also allows us to understand the role of $\gamma_{x,y}(t)$. Equation (20) suggests that the instantaneous rate combines the original rate with the relative likelihood of the evidence at T given y and x. If y is much more likely to lead to the final state, then the rates are biased toward y. Conversely, if y is unlikely to lead to the evidence the rate of transitions to it are lower. This observation also explains why the forward propagation of μ_x will reach the observed $\mu_x(T)$ even though we did not impose it explicitly. ∎

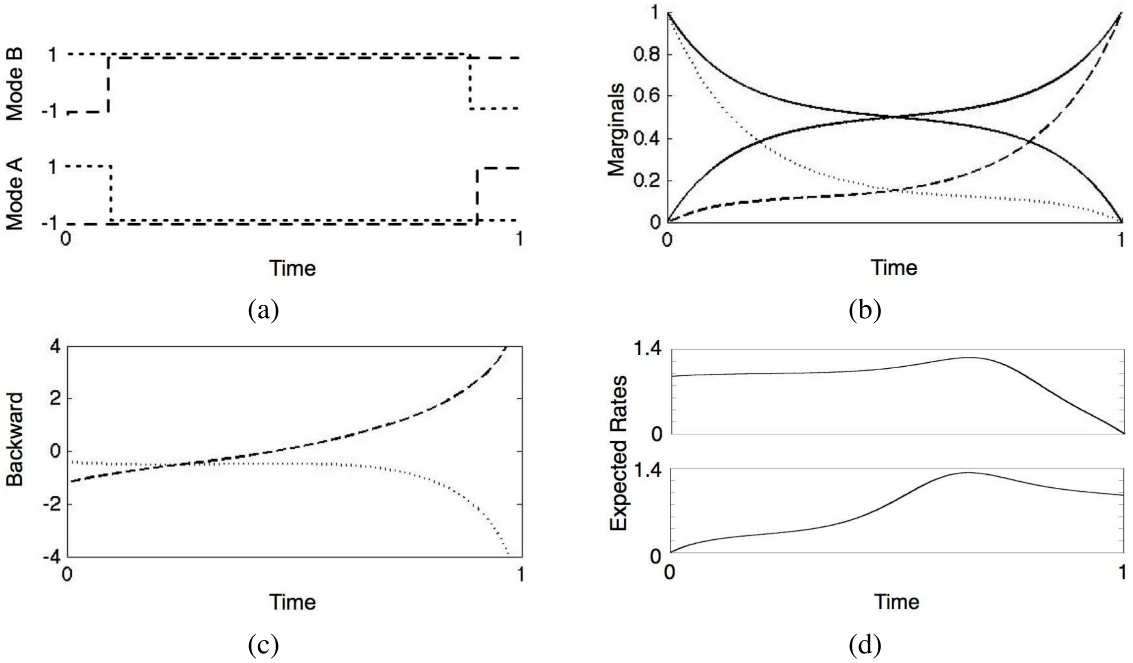
Example 3 Let us return to the two-component Ising chain in Example 1 with initial state $X_1^{(0)} = -1$ and $X_2^{(0)} = 1$, and a reversed state at the final time, $X_1^{(T)} = 1$ and $X_2^{(T)} = -1$. For a large value of β, this evidence is unlikely as at both end points the components are in a undesired configurations. The exact posterior is one that assigns higher probabilities to trajectories where one of the components switches relatively fast to match the other, and then toward the end of the interval, they separate to match the evidence. Since the model is symmetric, these trajectories are either ones in which both components are most of the time in state -1, or ones where both are most of the time in state 1 (Figure 3(a)). Due to symmetry, the marginal probability of each component is around 0.5 throughout most of the interval. The variational approximation cannot capture the dependency between the two components, and thus converges to one of two local maxima, corresponding to the two potential subsets of trajectories (Figure 3(b)). Examining the value of ρ^i, we see that close to the end of the interval they bias the instantaneous rates significantly. For example, as t approaches 1, $\rho^1_1(t)/\rho^1_{-1}(t)$ approaches infinity and so does the instantaneous rate $\gamma^1_{-1,1}(t)/\mu^1_{-1}(t)$, thereby forcing X_1 to switch to state 1 (Figure 3(c)).

This example also allows to examine the implications of modeling the posterior by inhomogeneous Markov processes. In principle, we might have used as an approximation Markov processes with homogeneous rates, and conditioned on the evidence. To examine whether our approximation behaves in this manner, we notice that in the single component case we have

$$q_{x,y} = \frac{\rho_x(t)\gamma_{x,y}(t)}{\rho_y(t)\mu_x(t)},$$

which should be constant.

Consider the analogous quantity in the multi-component case: $\tilde{q}^i_{x_i,y_i}(t)$, the geometric average of the rate of X_i, given the probability of parents state. Not surprisingly, this is exactly a mean field approximation, where the influence of interacting components is approximated by their average influence. Since the distribution of the parents (in the two-component system, the other component)



(a) (b) (c) (d)

Figure 3: Numerical results for the two-component Ising chain described in Example 3 where the first component starts in state -1 and ends at time $T = 1$ in state 1. The second component has the opposite behavior. *(a)* Two likely trajectories depicting the two modes of the model. *(b)* Exact (solid) and approximate (dashed/dotted) marginals $\mu^i_1(t)$. *(c)* The log ratio $\log \rho^i_1(t)/\rho^i_{-1}(t)$. *(d)* The expected rates $\tilde{q}^1_{1,-1}(t)$ and $\tilde{q}^1_{-1,1}(t)$ of component X_1 of the Ising chain in Example 1. We can notice that the averaged rates are highly non-constant, and so cannot be approximated well with a constant rate matrix.

changes in time, these rates change continuously, especially near the end of the time interval. This suggests that a piecewise homogeneous approximation cannot capture the dynamics without a loss in accuracy. As expected in a dynamic process, we can see in Figure 3(d) that the inhomogeneous transition rates are very erratic. In particular, the rates of X_1 spike at the transition point selected by the mean field approximation. This can be interpreted as putting most of the weight of the distribution on trajectories which transition from state -1 to 1 at that point. ∎

4.2 Optimization Procedure

If $\mathbb{Q}$ is irreducible, then $\rho^i_{x_i}$ and $\mu^i_{x_i}$ are non-zero throughout the open interval $(0,T)$. As a result, we can solve (17) to express $\gamma^i_{x_i,y_i}$ as a function of μ^i and ρ^i, thus eliminating it from (16) to get evolution equations solely in terms of μ^i and ρ^i. Abstracting the details, we obtain a set of ODEs of the form

$$\frac{d}{dt}\rho^i(t) = \alpha(\rho^i(t), \mu^{\setminus i}(t)) \qquad \rho^i(T) = \text{given},$$
$$\frac{d}{dt}\mu^i(t) = \beta(\mu^i(t), \rho^i(t), \mu^{\setminus i}(t)) \quad \mu^i(0) = \text{given}.$$

where α and β are defined by the right-hand side of the differential equations (16). Since the evolution of ρ^i does not depend on μ^i, we can integrate backward from time T to solve for ρ^i. Then, integrating forward from time 0, we compute μ^i. After performing a single iteration of backward-forward integration, we obtain a solution that satisfies the fixed-point equation (16) for the i'th component (this is not surprising once we have identified our procedure to be a variation of a standard forward-backward algorithm for a single component). Such a solution will be a local maximum of the functional w.r.t. to η^i (reaching a local minimum or a saddle point requires very specific initialization points).

This suggests that we can use the standard procedure of asynchronous updates, where we update each component in a round-robin fashion. Since each of these single-component updates converges in one backward-forward step, and since it reaches a local maximum, each step improves the value of the free energy over the previous one. As the free energy functional is bounded by the probability of the evidence, this procedure will always converge, and the rate of the free energy increase can be used to test for convergence.

Potentially, there can be many scheduling possibilities. In our implementation the update scheduling is simply random. A better choice would be to update the component which would maximally increase the value of the functional in that iteration. This idea is similar to the scheduling of Elidan et al. (2006), who approximate the change in the beliefs by bounding the *residuals* of the messages, which give an approximation of the benefit of updating each component.

Another issue is the initialization of this procedure. Since the iteration on the i'th component depends on $\mu^{\setminus i}$, we need to initialize μ by some legal assignment. To do so, we create a fictional rate matrix $\tilde{\mathbb{Q}}_i$ for each component and initialize μ^i to be the posterior of the process given the evidence $e_{i,0}$ and $e_{i,T}$. As a reasonable initial guess, we choose at random one of the conditional rates $\mathbb{Q}^{i|u_i}$ using some random assignment u_i to determine the fictional rate matrix.

The general optimization procedure is summarized in the following algorithm:

```
For each i, initialize $\mu^i$ using some legal marginal function.
while not converged do
    1. Pick a component $i \in \{1,\dots,D\}$.
    2. Update $\rho^i(t)$ by solving the $\rho^i$ backward differential equation in (16).
    3. Update $\mu^i(t)$ and $\gamma^i(t)$ by solving the $\mu^i$ forward differential equation in (16) and
       using the algebraic equation in (17).
end
```

Algorithm 1: Mean field approximation in continuous-time Bayesian networks

4.3 Exploiting Continuous-Time Representation

The continuous-time update equations allow us to use standard ODE methods with an adaptive step size (here we use the Runge-Kutta-Fehlberg (4,5) method). At the price of some overhead, these procedures automatically tune the trade-off between error and time granularity. Moreover, this overhead is usually negligible compared to the saving in computation time, because adaptive integration can be more efficient than *any* fixed step size integration by an order of magnitude (Press et al., 2007).

To further save computations, we note that while standard integration methods involve only initial boundary conditions at $t = 0$, the solution of μ^i is also known at $t = T$. Therefore, we stop the adaptive integration when $\mu^i(t) \approx \mu^i(T)$ and t is close enough to T. This modification reduces the number of computed points significantly because the derivative of μ^i tends to grow near the boundary, resulting in a smaller step size.

The adaptive solver selects different time points for the evaluation of each component. Therefore, updates of η^i require access to marginal density sets of neighboring components at time points that differ from their evaluation points. To allow efficient interpolation, we use a piecewise linear approximation of η whose boundary points are determined by the evaluation points that are chosen by the adaptive integrator.

5. Perspectives and Related Work

Variational approximations for different types of continuous-time processes have been recently proposed. Examples include systems with discrete hidden components (Opper and Sanguinetti, 2007); continuous-state processes (Archambeau et al., 2007); hybrid models involving both discrete and continuous-time components (Sanguinetti et al., 2009; Opper and Sanguinetti, 2010); and spatiotemporal processes (Ruttor and Opper, 2010; Dewar et al., 2010). All these models assume noisy observations in a finite number of time points. In this work we focus on structured discrete-state processes with noiseless evidence.

Our approach is motivated by results of Opper and Sanguinetti (2007) who developed a variational principle for a related model. Their model is similar to an HMM, in which the hidden chain is a continuous-time Markov process and there are (noisy) observations at discrete points along the process. They describe a variational principle and discuss the form of the functional when the approximation is a product of independent processes. There are two main differences between the setting of Opper and Sanguinetti and ours. First, we show how to exploit the structure of the target CTBN to reduce the complexity of the approximation. These simplifications imply that the update of the i'th process depends only on its Markov blanket in the CTBN, allowing us to develop efficient approximations for large models. Second, and more importantly, the structure of the evidence in our setting is quite different, as we assume deterministic evidence at the end of intervals. This setting typically leads to a posterior Markov process in which the instantaneous rates used by Opper and Sanguinetti diverge toward the end point—the rates of transition into the observed state go to infinity, leading to numerical problems at the end points. We circumvent this problem by using the marginal density representation which is much more stable numerically.

Taking the general perspective of Wainwright and Jordan (2008), the representation of the distribution uses the natural sufficient statistics. In the case of a continuous-time Markov process, the sufficient statistics are T_x, the time spent in state x, and $M_{x,y}$, the number of transitions from state x to y. In a discrete-time model, we can capture the statistics for every random variable. In a continuous-time model, however, we need to consider the time derivative of the statistics. Indeed, as shown in Section 3.3 we have

$$\frac{d}{dt}\mathbf{E}\left[T_x(t)\right] = \mu_x(t) \quad \text{and} \quad \frac{d}{dt}\mathbf{E}\left[M_{x,y}(t)\right] \quad = \gamma_{x,y}(t).$$

Thus, our marginal density sets η provide what we consider a natural formulation for variational approaches to continuous-time Markov processes.

Our presentation focused on evidence at two ends of an interval. Our formulation easily extends to deal with more elaborate types of evidence: (1) If we do not observe the initial state of the i'th component, we can set $\mu^i_x(0)$ to be the prior probability of $X^{(0)} = x$. Similarly, if we do not observe X_i at time T, we set $\rho^i_x(T) = 1$ as initial data for the backward step. (2) In a CTBN where one (or more) components are fully observed throughout some interval, we simply set μ^i for these components to be a distribution that assigns all the probability mass to the observed trajectory. Similarly, if we observe different components at different times, we may update each component on a different time interval. Consequently, maintaining for each component a marginal distribution μ^i throughout the interval of interest, we can update the other ones using their evidence patterns.

6. Evaluation on Ising Chains

To gain better insight into the quality of our procedure, we performed numerical tests on models that challenge the approximation. Specifically, we use Ising chains with the parameterization introduced in Example 1, where we explore regimes defined by the degree of coupling between the components (the parameter β) and the rate of transitions (the parameter τ). We evaluate the error in two ways. The first is by the difference between the true log-likelihood and our estimate. The second is by the average relative error in the estimate of different expected sufficient statistics defined by $\sum_j |\hat{\theta}_j - \theta_j|/\theta_j$, where θ_j is exact value of the j'th expected sufficient statistics and $\hat{\theta}_j$ is the approximation. To obtain a stable estimate the average is taken over all $\theta_j > 0.05 \max_{j'} \theta_{j'}$.

Applying our procedure on an Ising chain with 8 components, for which we can still perform exact inference, we evaluated the relative error for different choices of β and τ. The evidence in this experiment is $e_0 = \{+,+,+,+,+,+,-,-\}$, $T = 0.64$ and $e_T = \{-,-,-,+,+,+,+,+\}$. As shown in Figure 4(a), the error is larger when τ and β are large. In the case of a weak coupling (small β), the posterior is almost factored, and our approximation is accurate. In models with few transitions (small τ), most of the mass of the posterior is concentrated on a few canonical "types" of trajectories that can be captured by the approximation (as in Example 3). At high transition rates, the components tend to transition often, and in a coordinated manner, which leads to a posterior that is hard to approximate by a product distribution. Moreover, the resulting free energy landscape is rough with many local maxima. Examining the error in likelihood estimates (Figure 4(b),(c)) we see a similar trend.

Next, we examine the run time of our approximation when using fairly standard ODE solver with few optimizations and tunings. The run time is dominated by the time needed to perform the backward-forward integration when updating a single component, and by the number of such updates until convergence. Examining the run time for different choices of β and τ (Figure 5), we see that the run time of our procedure scales linearly with the number of components in the chain. The differences among the different curves suggest that the runtime is affected by the choice of parameters, which in turn affect the smoothness of the posterior density sets.

7. Evaluation on Branching Processes

The above-mentioned experimental results indicate that our approximation is accurate when reasoning about weakly-coupled components, or about time intervals involving few transitions (low transition rates). Unfortunately, in many domains we face strongly-coupled components. For example, we are interested in modeling the evolution of biological sequences (DNA, RNA, and proteins).

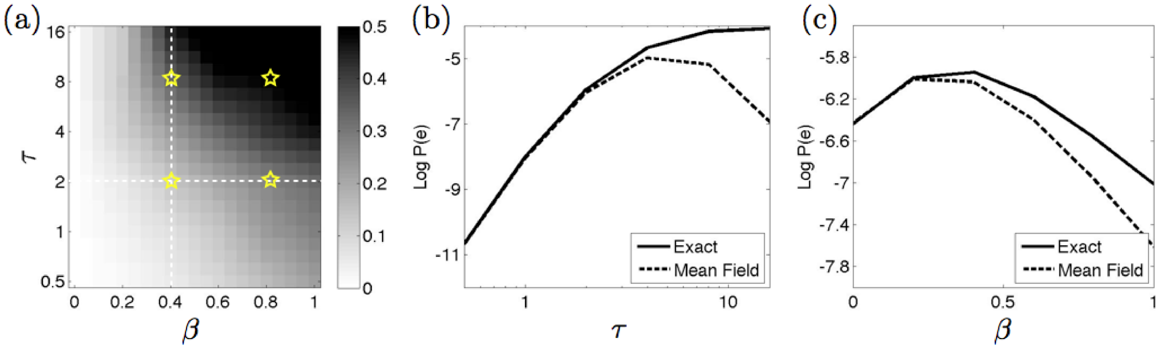


Figure 4: *(a)* Relative error as a function of the coupling parameter β (*x*-axis) and transition rates τ (*y*-axis) for an 8-component Ising chain. *(b)* Comparison of true vs. estimated likelihood as a function of the rate parameter τ. *(c)* Comparison of true vs. likelihood as a function of the coupling parameter β.

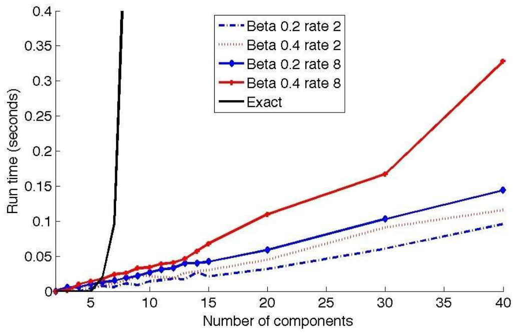


Figure 5: Evaluation of the run time of the approximation versus the run time of exact inference as a function of the number of components.

In such systems, we have a *phylogenetic tree* that represents the branching process that leads to current day sequences (see Figure 6).

It is common in sequence evolution to model this process as a continuous-time Markov process over a tree (Felsenstein, 2004). More precisely, the evolution along each branch is a standard continuous-time Markov process, and branching is modeled by a replication, after which each replica evolves independently along its sub-branch. Common applications are forced to assume that each character in the sequence evolves independently of the other.

In some situations, assuming an independent evolution of each character is highly unreasonable. Consider the evolution of an RNA sequence that folds onto itself to form a functional structure, as in Figure 7(a). This folding is mediated by complementary base-pairing (A-U, C-G, etc) that stabilizes the structure. During evolution, we expect to see compensatory mutations. That is, if a *A* changes into *C* then its based-paired *U* will change into a *G* soon thereafter. To capture such

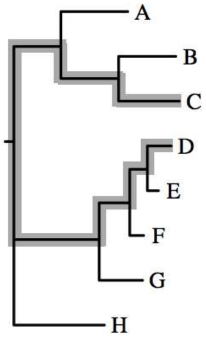


Figure 6: An example of a phylogenetic tree. Branch lengths denote time intervals between events. The interval used for the comparison with non-branching processes is highlighted.

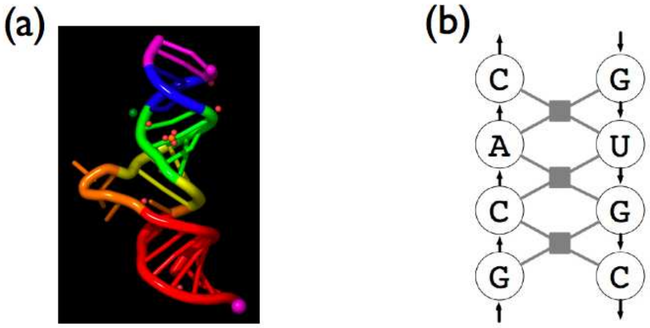


Figure 7: *(a)* Structure of an RNA molecule. The 3 dimensional structure dictates the dependencies between the different positions. *(b)* The form of the energy function for encoding RNA folding, superimposed on a fragment of a folded structure; each gray box denotes a term that involves four nucleotides.

coordinated changes, we need to consider the joint evolution of the different characters. In the case of RNA structure, the stability of the structure is determined by *stacking potentials* that measure the stability of two adjacent pairs of interacting nucleotides. Thus, if we consider a factor network to represent the energy of a fold, it will have structure as shown in Figure 7(b). We can convert this factor graph into a CTBN using procedures that consider the energy function as a fitness criteria in evolution (El-Hay et al., 2006; Yu and Thorne, 2006). Unfortunately, inference in such models suffers from computational blowup, and so the few studies that deal with it explicitly resort to sampling procedures (Yu and Thorne, 2006).

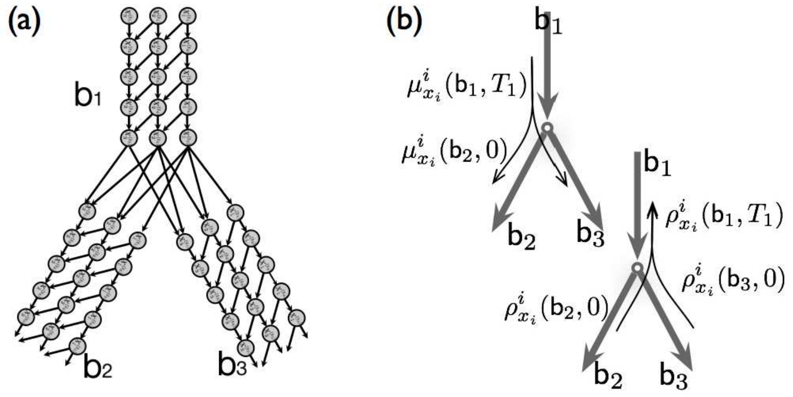

(a)
b1
b2
b3
(b)
b1
$\mu^i_{x_i}(\mathsf{b}_1, T_1)$
$\mu^i_{x_i}(\mathsf{b}_2, 0)$
b2
b3
b1
$\rho^i_{x_i}(\mathsf{b}_1, T_1)$
$\rho^i_{x_i}(\mathsf{b}_3, 0)$
$\rho^i_{x_i}(\mathsf{b}_2, 0)$
b2
b3

Figure 8: Structure of the branching process. *(a)* The discretized CTBN underlying the process in an intersection. *(b)* Illustration of the ODE updates on a directed tree, updating $\rho^i(t)$ backwards using (21) and $\mu^i(t)$ forwards using (22).

7.1 Representation

To consider phylogenetic trees, we should take a common approach in evolutionary analysis, in which inference of the tree topology and branch lengths is performed separately from inference of sequence dynamics. Thus, we need to extend our framework to deal with branching processes, where the branching points are fixed and known. In a linear-time model, we view the process as a map from $[0,T]$ into random variables $X^{(t)}$. In the case of a tree, we view the process as a map from a point $\mathsf{t} = \langle \mathsf{b}, t \rangle$ on a tree $\mathcal{T}$ (defined by branch b and the time t within it) into a random variable $X^{(\mathsf{t})}$. Similarly, we generalize the definition of the Markov-consistent density set η to include functions on trees. We define continuity of functions on trees in the obvious manner.

To gain intuition on this process we return to the discrete case, where our branching process can be viewed as a branching of the Dynamic Bayesian Network from one branch to two separate branches at the vertex, as in Figure 8(a).

7.2 Inference on Trees

The variational approximation on trees is thus similar to the one on intervals. Within each branch, we deal with the same update formulas as in linear time. We denote by $\mu^i_{x_i}(\mathsf{b}, t)$ and $\rho^i_{x_i}(\mathsf{b}, t)$ the messages computed on branch b at time t. The only changes occur at vertices, where we cannot use the Euler-Lagrange equations (Appendix E), therefore we must derive the propagation equations using a different method.

The following proposition establishes the update equations for the parameters $\mu^i(t)$ and $\rho^i(t)$ at the vertices, as depicted in Figure 8(b):

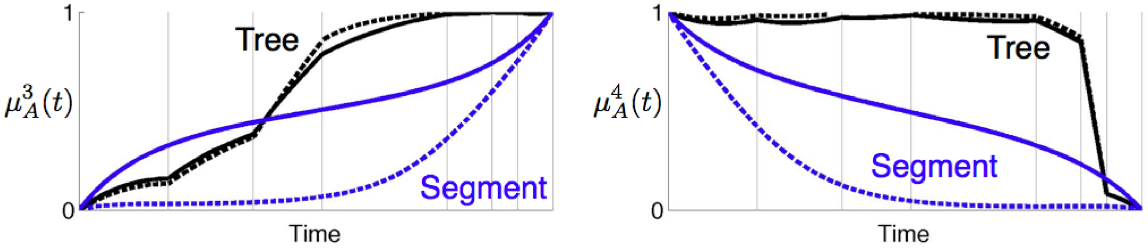


Figure 9: Comparison of exact vs. approximate inference along the highlighted path from C to D in the tree of Figure 6 with and without additional evidence at other leaves. In the latter case the problem is equivalent to inference on a linear segment. Exact marginals are shown in solid lines, whereas approximate marginals are in dashed lines. The horizontal gray lines indicate branch points along the path. Notice that evidence at the leaves result in discontinuities of the derivatives at such points. The two panels show two different components.

Proposition 7 *Given a vertex T with an incoming branch b_1 and two outgoing branches $\mathsf{b}_2, \mathsf{b}_3$. The following are the correct updates for our parameters $\mu^i_{x_i}(t)$ and $\rho^i_{x_i}(t)$:*

$$\rho^i_{x_i}(\mathsf{b}_1,T) = \rho^i_{x_i}(\mathsf{b}_2,0)\rho^i_{x_i}(\mathsf{b}_3,0), \tag{21}$$

$$\mu^i_{x_i}(\mathsf{b}_k,0) = \mu^i_{x_i}(\mathsf{b}_1,T) \qquad k=2,3. \tag{22}$$

Proof See Appendix G ■

Using Proposition 7 we can set the updates of the different parameters in the branching process according to (21–22). In the backward propagation of ρ^i, the value at the end of b_1 is the product of the values at the start of the two outgoing branches. This is the natural operation when we recall the interpretation of ρ^i as the probability of the downstream evidence given the current state (which is its exact meaning in a single component process): the downstream evidence of b_2 is independent of the downstream evidence of b_3, given the state of the process at the vertex $\langle \mathsf{b}_1, T\rangle$. The forward propagation of μ^i simply uses the value at the end of the incoming branch as initial value for the outgoing branches.

When switching to trees, we essentially increase the amount of evidence about intermediate states. Consider for example the tree of Figure 6 with an Ising chain model (as in the previous subsection). We can view the span from C to D as an interval with evidence at its end. When we add evidence at the tip of other branches we gain more information about intermediate points between C and D. Even though this evidence can represent evolution from these intermediate points, they do change our information state about them. To evaluate the impact of these changes on our approximation, we considered the tree of Figure 6, and compared it to inference in the backbone between C and D (Figure 4). Comparing the true marginal to the approximate one along the main backbone (see Figure 9) we see a major difference in the quality of the approximation. The evidence in the tree leads to a much tighter approximation of the marginal distribution. A more systematic comparison (Figure 10) demonstrates that the additional evidence reduces the magnitude of the error throughout the parameter space.

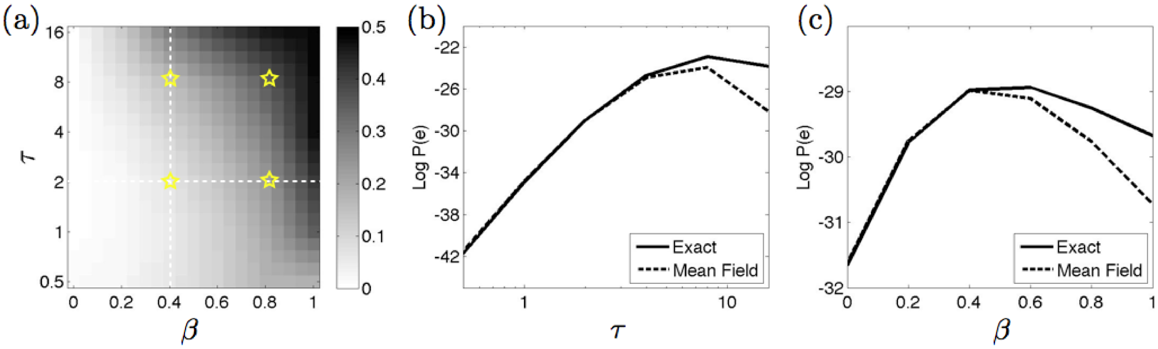


Figure 10: *(a)* Evaluation of the relative error in expected sufficient statistics for an Ising chain in branching-time; compare to Figure 4(a). *(b)*,*(c)* Evaluation of the estimated likelihood on a tree w.r.t. the rate τ and coupling β; compare to Figure 4(b),(c).

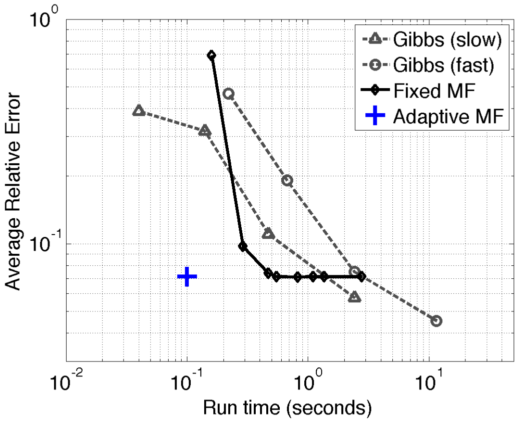


Figure 11: Evaluation of the run time vs. accuracy trade-off for several choices of parameters for mean field and Gibbs sampling on the branching process of Figure 6.

Similarly to mean-field, the Gibbs sampling procedure for CTBNs (El-Hay et al., 2008) can also be extended to deal with branching processes. Comparing our method to the Gibbs sampling procedure we see (Figure 11) that the faster mean field approach dominates the Gibbs procedure over short run times. However, as opposed to mean field, the Gibbs procedure is asymptotically unbiased, and with longer run times it ultimately prevails. This evaluation also shows that the adaptive integration procedure in our methods strikes a better trade-off than using a fixed time granularity integration.

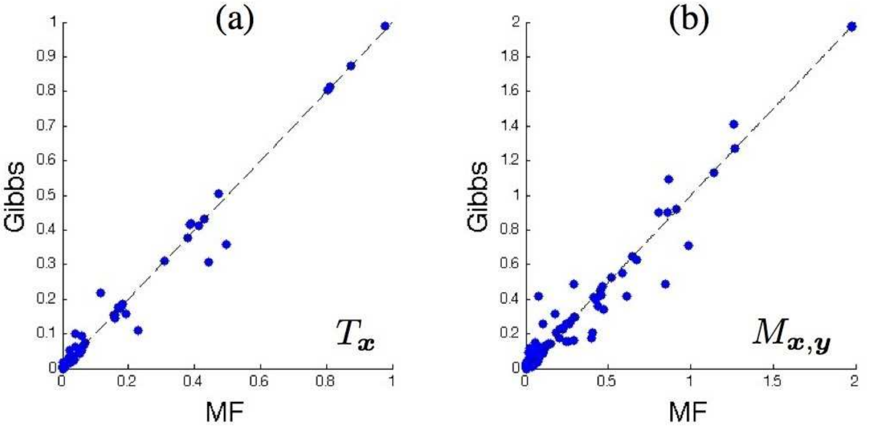


Figure 12: Comparison of estimates of expected sufficient statistics in the evolution of 18 interacting nucleotides, using a realistic model of RNA evolution. Each point is an expected value of: (a) residence time in a specific state of a component and its parents; (b) number of transition between two states. The x-axis is the estimate by the variational procedure, whereas the y-axis is the estimate by Gibbs sampling.

As a more demanding test, we applied our inference procedure to a model similar to the one introduced by Yu and Thorne (2006) for a stem of 18 interacting RNA nucleotides in 8 species in the phylogeny of Figure 6. In this model the transition rate between two sequences that differ in a single nucleotide depends on difference between their folding energy. Specifically, the transition rate from sequence x to sequence y is given by

$$q_{x,y} = 1.6\left(1 + e^{E_{\text{fold}}(y) - E_{\text{fold}}(x)}\right)^{-1}, \qquad |\delta(x,y)| = 1,$$

where E_{fold} is the folding energy of the sequence. This equation implies that transition rates are increasing monotonically with the reduction of the folding energy. Hence, this model tends to evolve into low energy states. The folding energy in turn is a sum of local stacking energies, involving quadruples of nucleotides as described by the factors in Figure 7. Denoting the subset of positions contained in each quadruple by D_k, the energy is

$$E_{\text{fold}}(x) = \sum_k E^k_{\text{fold}}(x|_{D_k}),$$

where $x|_{D_k}$ is the subset of nucleotides that belong factor k. This model is equivalent to a CTBN in which the parents of each components are the other components that share the same factors. This property follows from the fact that for any pair x and y, where $\delta(x,y) = \{i\}$, the difference between the energies of these two sequences depends only on the factors that contain i.

We compared our estimate of the expected sufficient statistics of this model to these obtained by the Gibbs sampling procedure. The Gibbs sampling estimates were chosen by running the procedure with an increasing computation time until there was no significant change in the results. The final estimates was obtained using 5000 burn-in rounds, 10000 number of samples and 100 rounds between two consecutive samples. The results, shown in Figure 12, demonstrate that over all the two approximate inference procedures are in good agreement about the value of the expected sufficient statistics.

8. Discussion

In this paper we formulate a general variational principle for continuous-time Markov processes (by reformulating and extending the one proposed by Opper and Sanguinetti, 2007), and use it to derive an efficient procedure for inference in CTBNs. In this mean field approximation, we use a product of independent inhomogeneous processes to approximate the multi-component posterior.

Our procedure enjoys the same benefits encountered in discrete-time mean field procedure (Jordan et al., 1999): it provides a lower-bound on the likelihood of the evidence and its run time scales linearly with the number of components. Using asynchronous updates it is guaranteed to converge, and the approximation represents a consistent joint distribution. It also suffers from expected shortcomings: the functional has multiple local maxima, it cannot capture complex interactions in the posterior (Example 3). By using a time-inhomogeneous representation our approximation does capture complex patterns in the temporal progression of the marginal distribution of each component. Importantly, the continuous-time parameterization enables straightforward implementation using standard ODE integration packages that automatically tune the trade-off between time granularity and approximation quality. We show how it is extended to perform inference on phylogenetic trees, where the posterior distribution is directly affected by several evidence points, and show that it provides fairly accurate answers in the context of a real application (Figure 12).

A key development is the introduction of marginal density sets. Using this representation we reformulate and extend the variational principle proposed by Opper and Sanguinetti (2007) , which incorporates a different inhomogeneous representation. This modification allows handling direct evidence of the state of the process, as in the case of CTBNs, while keeping the representation of the approximation bounded. The extension of this principle to CTBNs follows by exploiting their networks structure. This adaptation of continuously inhomogeneous representations to CTBNs increases the flexibility of the approximation relative to the piecewise homogeneous representation of Saria et al. (2007) and, somewhat surprisingly, also significantly simplifies the resulting formulation.

The proposed representation is natural in the sense that its functions are the time-derivatives of the expected sufficient statistics that we are willing to evaluate. Hence, once finding the optimal value of the lower bound, calculating these expectations is straightforward. This representation is analogous to mean parameters which have proved powerful in variational approximations of exponential families over finite random variable sets (Wainwright and Jordan, 2008).

We believe that even in cases where evidence is indirect and noisy, the marginal density representation should comprise smoother functions than posterior rates. Intuitively, in the presence of a noisy observation the posterior probability of some state x can be very small. In such cases, the posterior transition rate form x into a state that better explains the observation might tend to a large quantity. This reasoning suggests that marginal density representations should be better handled by adaptive numerical integration algorithms. An interesting direction would be to test this conjecture empirically.

A possible extension is using our variational procedure to generate the initial distribution for the Gibbs sampling procedure and thus skip the initial burn-in phase and produce accurate samples. Another attractive aspect of this new variational approximation is its potential use for learning model parameters from data. It can be easily combined with the EM procedure for CTBNs (Nodelman et al., 2005a) to obtain a Variational-EM procedure for CTBNs, which monotonically increases the likelihood by alternating between steps that improve the approximation η (the updates discussed here) and steps that improve the model parameters θ (an M-step Nodelman et al., 2005a). Finally, marginal density sets are a particularly suitable representation for adapting richer representations such as Bethe, Kikuchi and convex approximations to non-homogeneous versions (El-Hay et al., 2010). Further work in that direction should allow bridging the gap in the wealth of inference techniques between finite domain models and continuous-time models.

Acknowledgments

We thank Ofer Meshi, Menachem Fromer and the anonymous reviewers for helpful remarks on previous versions of the manuscript. This research was supported in part by a grant from the Israel Science Foundation. Tal El-Hay is supported by the Eshkol fellowship from the Israeli Ministry of Science.

Appendix A. The Relation Between CTBNs and DBNs

In this section we show that the DBN construction of Equations (6-7) is such that as h approaches 0, the distribution P_h approaches Pr. To show this, it suffice to show that

$$\lim_{h\to 0} \frac{P_h(X^{(t_{k+1})} = y | X^{(t_k)} = x) - \mathbf{1}_{x=y}}{h} = q_{x,y} \quad .$$

We ensured this condition holds component-wise, and now need to show that this leads to global consistency.

Plugging Equation (7) into Equation (6), the transition probability of the DBN is

$$P_h(X^{(t_{k+1})} = y | X^{(t_k)} = x) = \prod_i \left(\mathbf{1}_{x_i=y_i} + q^{i|\mathbf{Pa}_i}_{x_i,y_i|u_i} \cdot h \right) \quad .$$

Since we consider the limit as h approaches 0, any term that involves h^d with $d > 1$ is irrelevant. And thus, we can limit our attention to the constant terms and terms linear in h. Expanding the product gives

$$P_h(X^{(t_{k+1})} = y | X^{(t_k)} = x) = \prod_i \mathbf{1}_{x_i=y_i} + \sum_i q^{i|\mathbf{Pa}_i}_{x_i,y_i|u_i} \cdot h \prod_{j\neq i} \mathbf{1}_{x_j=y_j} + o(h) \quad .$$

Now, $\prod_i \mathbf{1}_{x_i=y_i} = \mathbf{1}_{x=y}$. Moreover, it is easy to verify that

$$q_{x,y} = \sum_i q^{i|\mathbf{Pa}_i}_{x_i,y_i|u_i} \prod_{j\neq i} \mathbf{1}_{x_j=y_j} \quad .$$

Thus,

$$P_h(X^{(t_{k+1})} = y | X^{(t_k)} = x) = \mathbf{1}_{x=y} + q_{x,y}h + o(h),$$

proving the required condition.

Appendix B. Marginal Density Sets and Markov Processes - Proof of Lemma 2

Proof Given η, we define the *inhomogeneous rate matrix* $\mathbb{R}(t)$ as in Equation (11). $\mathbb{R}(t)$ is a valid rate matrix because its off-diagonals are non-negative as they are the quotient of two non-negative functions, and because applying the requirement on $\gamma_{x,x}(t)$ in Definition 1

$$r_{x,x}(t) = \frac{\gamma_{x,x}(t)}{\mu_x(t)} = -\frac{\sum_{y\neq x}\gamma_{x,y}(t)}{\mu_x(t)} = -\sum_{y\neq x} r_{x,y}(t) \ ,$$

we see that $\mathbb{R}(t)$'s diagonals are negative and the rows sum to 0. We can use these rates with the initial value $\mu_x(0)$ to construct the Markov process P_η from the forward master equation

$$\frac{d}{dt}P_\eta(X^{(t)} = x) = \sum_y P_\eta(X^{(t)} = y) r_{y,x}(t) \ ,$$

and

$$P_\eta(X^{(0)}) = \mu(0) \ .$$

To conclude the proof we show that P_η and the marginal density set satisfy (10). First, from Definition 1 it follows that $\mu(t)$ is the solution to the master equation of $P_\eta(X^{(t)})$, because the initial values match at $t = 0$ and the time-derivatives of the two functions are identical. Thus

$$P_\eta(X^{(t)} = x) = \mu_x(t) \ .$$

Next, the equivalence of the joint probability densities can be proved:

$$\begin{aligned}
\lim_{h\to 0}\frac{\Pr(X^{(t)} = x, X^{(t+h)} = y)}{h} &= \lim_{h\to 0}\frac{\mu_x(t)\Pr(X^{(t+h)} = y|\Pr(X^{(t)} = x)}{h} \\
&= \lim_{h\to 0}\frac{\mu_x(t) r_{x,y}(t) h}{h} \\
&= \mu_x(t) r_{x,y}(t) \ .
\end{aligned}$$

From the definition of $r_{x,y}(t)$ and the fact that $\gamma_{x,y}(t) = 0$ whenever $\mu_x(t) = 0$, it follows that $\mu_x(t) r_{x,y}(t)$ is exactly $\gamma_{x,y}(t)$ ■

Appendix C. Expectations in Inhomogeneous Processes

This section includes the proofs of the lemmas used in the proof of the variational lower bound theorem.

C.1 Expectations of Functions of States - Proof of Lemma 4

Proof Changing the order of integration we obtain

$$\mathbf{E}_{f_\eta}\left[\int_0^T \psi(\sigma(t),t)dt\right] \equiv \int_\Sigma f_\eta(\sigma)\int_0^T \psi(\sigma(t),t)dtd\sigma = \int_0^T \int_\Sigma f_\eta(\sigma)\cdot\psi(\sigma(t),t)\,d\sigma dt \ .$$

For each $t \in T$ we decompose the inner integral according to possible states at that time:

$$\begin{aligned}\int_\Sigma f_\eta(\sigma)\cdot\psi(\sigma(t),t)\,d\sigma &= \sum_x \int_\Sigma f_\eta(\sigma)\cdot \mathbf{1}_{\sigma(t)=x}\cdot\psi(x,t)\,d\sigma\\ &= \sum_x \psi(x,t)\int_\Sigma f_\eta(\sigma)\cdot \mathbf{1}_{\sigma(t)=x}\,d\sigma\\ &= \sum_x \psi(x,t)\mu_x(t)\ .\end{aligned}$$

■

C.2 Expectations of Functions of Transitions - Proof of Lemma 5

Proof Given a trajectory σ there exists a small enough $h>0$ such that for every transition and for every $t\in(t_k-h,t_k)$ we have $\sigma(t)=x_{k-1}$ and $\sigma(t+h)=x_k$. In that case we can rewrite the sum in the expectation term as

$$\begin{aligned}\sum_{k=1}^{K^\sigma}\psi(x^\sigma_{k-1},x^\sigma_k,t^\sigma_k) &= \sum_{k=1}^{K^\sigma}\frac{1}{h}\int_{t_k-h}^{t_k}\psi(\sigma(t),\sigma(t+h),t)dt+\frac{o(h)}{h}\\ &= \frac{1}{h}\int_0^{T-h}\psi(\sigma(t),\sigma(t+h),t)dt+\frac{o(h)}{h}\ ,\end{aligned}$$

where the first equality follows from continuity and the second one from the requirement that $\psi(x,x,t)=0$. Taking the limit $h\to 0$ and using this requirement again gives

$$\sum_{k=1}^{K^\sigma}\psi(x^\sigma_{k-1},x^\sigma_k,t^\sigma_k)=\frac{d}{ds}\left[\int_0^T\psi(\sigma(t),\sigma(t+s),t)dt\right]_{s=0}\ .$$

Taking expectation we obtain

$$\begin{aligned}&\int_\Sigma f(\sigma)\frac{d}{ds}\left[\int_0^T\psi(\sigma(t),\sigma(t+s),t)dt\right]_{s=0}d\sigma\\ &= \int_\Sigma f(\sigma)\frac{d}{ds}\left[\int_0^T\sum_x\sum_{y\neq x}\psi(x,y,t)\,\mathbf{1}_{\sigma(t)=x}\mathbf{1}_{\sigma(t+s)=y}\,dt\right]_{s=0}d\sigma\\ &= \frac{d}{ds}\left[\int_0^T\sum_x\sum_{y\neq x}\psi(x,y,t)\int_\Sigma f(\sigma)\,\mathbf{1}_{\sigma(t)=x}\mathbf{1}_{\sigma(t+s)=y}d\sigma\,dt\right]_{s=0}\ .\end{aligned}$$

The inner integral in the last term is a joint probability

$$\int_\Sigma f(\sigma)\,\mathbf{1}_{\sigma(t)=x}\mathbf{1}_{\sigma(t+s)=y}d\sigma=\Pr(X^{(t)}=x,X^{(t+s)}=y)\ .$$

Switching the order of integration and differentiation and using

$$\frac{d}{ds}\Pr(X^{(t)}=x,X^{(t+s)}=y)\bigg|_{s=0}=\gamma_{xy}(t),\quad x\neq y,$$

gives the desired result. ■

Appendix D. Proof of the Factored Representation of the Energy Functional

Proof We begin with the definition of the average energy

$$\begin{aligned}
\mathcal{E}(\eta;\mathbb{Q}) &= \int_0^T \sum_x \left[\mu_x(t) q_{x,x} + \sum_{y\neq x} \gamma_{x,y}(t) \ln q_{x,y}\right] dt \\
&= \int_0^T \sum_x \left[\mu_x(t) q_{x,x} + \sum_i \sum_{y_i\neq x_i} \gamma^i_{x_i,y_i}(t) \mu^{\backslash i}(t) \ln q_{x,y}\right] dt \ .
\end{aligned}$$

where the equality stems from the observation that the only states y that may have $\gamma_{x,y}(t) > 0$, are those with $\delta(x,y) \leq 1$ (all the rest are 0). Thus, the enumeration over all possible states collapses into an enumeration over all components i and all states $y_i \neq x_i$. Due to the fact that we are only considering transitions in single components, we may replace the global joint density $\gamma_{x,y}$ with $\gamma^i_{x_i,y_i} \cdot \mu^{\backslash i}(t)$, as per definition.

Using (5), we can decompose the transition rates $q_{x,x}$ and $q_{x,y}$ to get

$$\begin{aligned}
\mathcal{E}(\eta;\mathbb{Q}) &= \sum_i \int_0^T \sum_x \left[\mu_x(t) q_{x_i,x_i|u_i} + \sum_{y_i\neq x_i} \gamma^i_{x_i,y_i}(t) \mu^{\backslash i}(t) \ln q_{x_i,y_i|u_i}\right] dt \\
&= \sum_i \int_0^T \sum_{x_i} \left[\mu^i_{x_i}(t) \sum_{x\backslash i} \mu^{\backslash i}_{x\backslash i}(t) q_{x_i,x_i|u_i} + \sum_{y_i\neq x_i} \gamma^i_{x_i,y_i}(t) \mu^{\backslash i}_{x\backslash i}(t) \ln q_{x_i,y_i|u_i}\right] dt \ \ .
\end{aligned}$$

To get to the last equality we use the factorization of $\mu(t)$ as a product of $\mu^i(t)$ with $\mu^{\backslash i}(t)$ and the reordering of the summation. Next we simply write the previous sum as an expectation over $X \setminus i$

$$\mathcal{E}(\eta;\mathbb{Q}) = \sum_i \int_0^T \sum_{x_i} \mu^i_{x_i}(t) \mathbf{E}_{\mu^{\backslash i}(t)} \left[q_{x_i,x_i|U_i}\right] + \sum_i \int_0^T \sum_{y_i\neq x_i} \gamma^i_{x_i,y_i}(t) \mathbf{E}_{\mu^{\backslash i}(t)} \left[\ln q_{x_i,y_i|U_i}\right] dt \ ,$$

which concludes the proof.

Turning to the entropy-like term we have:

$$\begin{aligned}
\mathcal{H}(\eta) &= \int_0^T \sum_x \sum_{y\neq x} \gamma_{x,y}(t)[1 + \ln\mu_x(t) - \ln\gamma_{x,y}(t)]dt \\
&= \sum_i \int_0^T \sum_x \sum_{y_i\neq x_i} \mu^{\backslash i}(t)\gamma_{x_i,y_i}(t)[1 + \sum_i \ln\mu^i_{x_i}(t) - \ln\gamma_{x_i,y_i}(t) - \sum_{j\neq i} \ln\mu_{x_j}(t)]dt \\
&= \sum_i \int_0^T \sum_{x_i} \sum_{y_i\neq x_i} \gamma_{x_i,y_i}(t)[1 + \ln\mu^i_{x_i}(t) - \ln\gamma_{x_i,y_i}(t)]dt \\
&= \sum_i \mathcal{H}(\eta^i) \ ,
\end{aligned}$$

where, the first equality is definition of $\mathcal{H}$. The second one follows from the definition of the factored density set. The third one is obtained by algebraic manipulation and the last one is again the definition of $\mathcal{H}$. ■

Appendix E. Euler-Lagrange Equations

The problem of finding the fixed points of *functionals* whose arguments are continuous functions comes from the field of *Calculus of variations*. We briefly review the usage Euler-Lagrange equation for solving optimization problems over functionals. Additional information can be found in Gelfand and Fomin (1963).

A functional is a mapping from a vector space to its underlying field. In our case the functional is the Lagrangian introduced in Section 4, which is an integral over real-valued functions, and the underlying field is the real numbers.

Given a functional over a normed space of continuously differentiable real functions of the form

$$I[y] = \int_a^b f(t, y(t), y'(t))dt$$

where $y'(t)$ is the time-derivative of the function $y(t)$, we would like to find a function $y(t)$ that minimizes (or in our case maximizes) the functional subject to $y(a) = y_a$ and $y(b) = y_b$. In the simplest case, when there are no additional constraints, a necessary condition for y to be a local optimum is that y is a *stationary point*. Roughly, a stationary point is a function y, where $I[y]$ is insensitive to small variations in y. That is, given a function $h(t)$ where $h(a) = 0$ and $h(b) = 0$, the change of the functional $I[y+h] - I[y]$ is small relative to the norm of h. For $y(t)$ to be a stationary point, it must satisfy the *Euler-Lagrange* equations (Gelfand and Fomin, 1963)

$$\frac{\partial}{\partial y} f(t, y(t), y'(t)) - \frac{d}{dt}\left(\frac{\partial}{\partial y'} f(t, y(t), y'(t))\right) = 0 \ . \tag{23}$$

In this paper we have additional constraints describing the time derivative of μ. The generalization of the Euler-Lagrange equations to that case is straightforward. Denoting the subsidiary constraints by $g(t, y(t), y'(t)) = 0$, we simply replace $f(t, y, y')$ by $f(t, y, y') - \lambda(t)g(t, y, y')$ in Equation 23.

An example for the use of this equation is in the following proof.

Appendix F. Stationary Points of the Lagrangian - Proof of Theorem 6

Proof For convenience, we begin by rewriting the Lagrangian in explicit form: $\mathcal{L} = \int_0^T f(y(t), y'(t))dt$ where $y(t) = \langle \mu(t), \gamma(t), \lambda(t) \rangle$ is a concatenation of the parameters and Lagrange multiplier and

$$\begin{aligned} f(y(t), y'(t)) \quad = \quad & \sum_{i=1}^{D} \sum_{x_i} \Bigg[\mu^i_{x_i}(t) \mathbf{E}_{\mu^{\setminus i}(t)} \left[q_{x_i, x_i|U_i} \right] + \sum_{y_i \neq x_i} \gamma^i_{x_i, y_i}(t) \mathbf{E}_{\mu^{\setminus i}(t)} \left[\ln q_{x_i, y_i|U_i} \right] \\ & + \sum_{y_i \neq x_i} \gamma_{x_i y_i} \left[1 + \ln \mu^i_{x_i}(t) - \ln \gamma^i_{x_i y_i}(t) \right] - \lambda^i_{x_i}(t) \left(\frac{d}{dt} \mu^i_{x_i}(t) - \sum_{y_i} \gamma^i_{x_i y_i}(t) \right) \Bigg] \ . \end{aligned}$$

The Euler-Lagrange equations of the Lagrangian define its stationary points w.r.t. the parameters of each component $\mu^i(t)$, $\gamma^i(t)$ and $\lambda^i(t)$.

First, we take the partial derivatives of f w.r.t to $\mu^i_{x_i}(t)$ as well as $\frac{d}{dt}\mu^i_{x_i}(t)$ and plug them into Equation 23. We start by handling the energy terms. These terms involve expectations in the form

$\mathbf{E}_{\mu\setminus j(t)}\left[g(U_j)\right] = \sum_{u_j} \mu_{u_j}(t) g(u_j)$. The parameter $\mu^i_{x_i}(t)$ appears in these terms only when i is a parent of j and u_j is consistent with x_i. In that case $\frac{\partial}{\partial \mu^i_{x_i}} \mu_{u_j} = \mu_{u_j}/\mu^i_{x_i}$. Thus,

$$\frac{\partial}{\partial \mu^i_{x_i}} \mathbf{E}_{\mu\setminus j(t)}\left[g(U_j)\right] = \mathbf{E}_{\mu\setminus j(t)}\left[g(U_j) \mid x_i\right] \cdot \delta_{j\in Children_i}$$

Recalling the definitions of the averaged rates

$$\bar{q}^{\,i}_{x_i,x_i|x_j}(t) = \mathbf{E}_{\mu\setminus i(t)}\left[q^{i|\mathbf{Pa}_i}_{x_i,x_i|U_i} \mid x_j\right]$$

and

$$\tilde{q}^{i}_{x_i,y_i|x_j}(t) = \exp\left\{\mathbf{E}_{\mu\setminus i(t)}\left[\ln q^{i|\mathbf{Pa}_i}_{x_i,y_i|U_i} \mid x_j\right]\right\}$$

we obtain

$$\frac{\partial}{\partial \mu^i_{x_j}} \mathbf{E}_{\mu\setminus j(t)}\left[q^{j}_{x_j,x_j|U_j}\right] = \delta_{j\in Children_i} \bar{q}^{\,j}_{x_j,x_j|x_i}(t)$$

and

$$\frac{\partial}{\partial \mu^i_{x_j}} \mathbf{E}_{\mu\setminus j(t)}\left[\ln q^{j}_{x_j,x_j|U_j}\right] = \delta_{j\in Children_i} \ln \tilde{q}^{j}_{x_j,x_j|x_i}(t).$$

Therefore the derivative of the sum over $j \neq i$ of the energy terms is

$$\psi^i_{x_i}(t) \equiv \sum_{j\in Children_i} \sum_{x_j} \left[\mu^j_{x_j}(t)\bar{q}^{\,j}_{x_j,x_j|x_i}(t) + \sum_{x_j\neq y_j} \gamma^j_{x_j,y_j}(t) \ln \tilde{q}^{j}_{x_j,y_j|x_i}(t)\right] \quad .$$

Additionally, the derivative of the energy term for $j = i$ is $\bar{q}^{\,i}_{x_i,x_i}(t) \equiv \mathbf{E}_{\mu\setminus i(t)}\left[q_{x_i,x_i|U_i}\right]$. Next, the derivative of the entropy term is $\gamma^i_{x_i,x_i}(t)/\mu^i_{x_i}(t)$. Finally, the derivative of f with respect to $\frac{d}{dt}\mu^i_{xi}(t)$ is $-\lambda^i_{x_i}(t)$. Plugging in these derivatives into Equation (23) we obtain

$$\bar{q}^{\,i}_{x_i,x_i}(t) + \psi^i_{x_i}(t) - \frac{\gamma^i_{x_i,x_i}}{\mu^i_{x_i}(t)} + \frac{d}{dt}\lambda^i_{x_i}(t) = 0 \quad . \tag{24}$$

Next, the derivative w.r.t. $\gamma^i_{x_i,y_i}(t)$ gives us

$$\ln \mu^i_{x_i}(t) + \ln \tilde{q}^i_{x_i,y_i}(t) - \ln \gamma^i_{x_i,y_i}(t) + \lambda^i_{y_i}(t) - \lambda^i_{x_i}(t) = 0 \quad . \tag{25}$$

Denoting $\rho^i_{x_i}(t) = \exp\{\lambda^i_{x_i}(t)\}$, Equation (25) becomes

$$\gamma^i_{x_i,y_i}(t) = \mu^i_{x_i}(t)\tilde{q}^i_{x_i,y_i}(t)\frac{\rho^i_{y_i}(t)}{\rho^i_{x_i}(t)} \quad ,$$

which is the algebraic equation of γ. Using this result and the definition of $\gamma^i_{x_i,x_i}$ we have

$$\gamma^i_{x_i,x_i}(t) = -\sum_{y_i\neq x_i} \gamma^i_{x_i,y_i}(t) = -\mu^i_{x_i}(t) \sum_{x_i,y_i} \tilde{q}^i_{x_i,y_i}(t)\frac{\rho^i_{y_i}(t)}{\rho^i_{x_i}(t)}.$$

Plugging this equality into (24) and using the identity $\frac{d}{dt}\rho^i_{x_i}(t) = \frac{d}{dt}\lambda^i_{x_i}(t)\rho^i_{x_i}(t)$ gives

$$\frac{d}{dt}\rho^i_{x_i}(t) = -\rho^i_{x_i}(t)\left(\bar{q}^{\,i}_{x_i,x_i}(t) + \psi^i_{x_i}(t)\right) - \sum_{y_i\neq x_i} \tilde{q}^i_{x_i,y_i}\rho^i_{y_i}(t) \quad .$$

Thus the stationary point of the Lagrangian matches the updates of (16–17). ■

Appendix G. Proof of Proposition 7

Proof We denote the time at the vertex $t_0 = (\mathsf{b}_1, T)$, the time just before as $t_1 = (\mathsf{b}_1, T-h)$ and the times just after it on each branch $t_2 = (\mathsf{b}_2, h)$ and $t_3 = (\mathsf{b}_3, h)$, as in Figure 13.

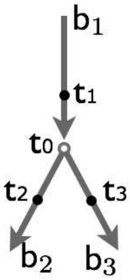


Figure 13: Branching process with discretization points of Lemma 7.

The marginals $\mu^i_{x_i}(\mathsf{b}_1, t)$ are continuous, as they are derived from the forward differential equation. To derive the propagation formula for the $\rho^i_{x_i}(t)$ requires additional care. The $\rho^i_{x_i}(t)$ have been derived from the constraints on the time-derivative of $\mu^i_{x_i}(t)$. In a vertex this constraint is threefold, as we now have the constraints on b_1

$$\frac{\mu^i_{x_i}(t_0) - \mu^i_{x_i}(t_1)}{h} = \sum_{y_i} \gamma^i_{x_i,y_i}(t_1)$$

and those on the other branches b_k for $k = 2, 3$

$$\frac{\mu^i_{x_i}(t_k) - \mu^i_{x_i}(t_0)}{h} = \sum_{y_i} \gamma^i_{x_i,y_i}(t_0) \quad .$$

The integrand of the Lagrangian corresponding to point t_0 is

$$\begin{aligned} \mathcal{L}_{|t_0} &= \tilde{\mathcal{F}}(\eta; \mathbb{Q})_{|t_0} + \lambda^0(t_1) \left(\frac{\mu^i_{x_i}(t_0) - \mu^i_{x_i}(t_1)}{h} - \sum_{y_i} \gamma^i_{x_i,y_i}(t_1) \right) \\ &- \sum_{k=2,3} \lambda^k(t_0) \left(\frac{\mu^i_{x_i}(t_k) - \mu^i_{x_i}(t_0)}{h} - \sum_{y_i} \gamma^i_{x_i,y_i}(t_0) \right) \quad , \end{aligned}$$

as this is the only integrand which involves $\mu_{x_i}(t_0)$, the derivative of the Lagrangian collapses into

$$\begin{aligned} \frac{\partial}{\partial \mu^i_{x_i}(t_0)} \mathcal{L} &= \frac{\partial}{\partial \mu^i_{x_i}(t_0)} \mathcal{L}_{|t_0} \\ &= \frac{\lambda^0(t_1)}{h} - \left(\frac{\lambda^2(t_0)}{h} + \frac{\lambda^3(t_0)}{h} \right) + \frac{\partial}{\partial \mu^i_{x_i}(t_0)} \tilde{\mathcal{F}}(\eta; \mathbb{Q})_{|t_0} = 0 \quad . \end{aligned}$$

Rearranging the previous equation and multiplying by h, we get

$$\lambda^0(t_1) = \lambda^2(t_0) + \lambda^3(t_0) + \frac{\partial}{\partial \mu^i_{x_i}(t_0)} \tilde{\mathcal{F}}(\eta; \mathbb{Q})_{|t_0} h \quad .$$

Looking at (24) we can see that as t_0 is not a leaf, and thus $\mu^i_{x_i}(t_0) > 0$ and the derivative of the functional cannot diverge. Therefore, as $h \to 0$ this term vanishes and we are left with

$$\lambda^0(t_1) = \lambda^2(t_0) + \lambda^3(t_0)$$

which after taking exponents gives (21). ■

References

C. Archambeau, M. Opper, Y. Shen, D. Cornford, and J. Shawe-Taylor. Variational inference for diffusion processes. In *Advances in Neural Information Processing Systems 20*. MIT Press, 2007.

K. L. Chung. *Markov Chains with Stationary Transition Probabilities*. Springer Verlag, Berlin, 1960.

T. Dean and K. Kanazawa. A model for reasoning about persistence and causation. *Comput. Intell.*, 5(3):142–150, 1989.

M. Dewar, V. Kadirkamanathan, M. Opper, and G. Sanguinetti. Parameter estimation and inference for stochastic reaction-diffusion systems: application to morphogenesis in d. melanogaster. *BMC Systems Biology*, 4(1):21, 2010.

T. El-Hay, N. Friedman, D. Koller, and R. Kupferman. Continuous time markov networks. In *Proc. Twenty-second Conference on Uncertainty in Artificial Intelligence (UAI)*, 2006.

T. El-Hay, N. Friedman, and R. Kupferman. Gibbs sampling in factorized continuous-time markov processes. In *Proc. Twenty-fourth Conference on Uncertainty in Artificial Intelligence (UAI)*, 2008.

T. El-Hay, I. Cohn, N. Friedman, and R. Kupferman. Continuous-time belief propagation. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, 2010.

G. Elidan, I. Mcgraw, and D. Koller. Residual belief propagation: informed scheduling for asynchronous message passing. In *Proc. Twenty-second Conference on Uncertainty in Artificial Intelligence (UAI)*, 2006.

Y. Fan and C. R. Shelton. Sampling for approximate inference in continuous time Bayesian networks. In *Tenth International Symposium on Artificial Intelligence and Mathematics*, 2008.

Y. Fan and C. R. Shelton. Learning continuous-time social network dynamics. In *Proc. Twenty-fifth Conference on Uncertainty in Artificial Intelligence (UAI)*, 2009.

J. Felsenstein. *Inferring Phylogenies*. Sinauer, 2004.

C. W. Gardiner. *Handbook of Stochastic Methods*. Springer-Verlag, New-York, third edition, 2004.

I. M. Gelfand and S. V. Fomin. *Calculus of Variations*. Prentice-Hall, 1963.

K. Gopalratnam, H. Kautz, and D. S. Weld. Extending continuous time bayesian networks. In *Proceedings of the 20th National Conference on Artificial Intelligence (AAAI)*, pages 981–986. AAAI Press, 2005.

M. I. Jordan, Z. Ghahramani, T. Jaakkola, and L. K. Saul. An introduction to variational approximations methods for graphical models. In M. I. Jordan, editor, *Learning in Graphical Models*. MIT Press, Cambridge MA, 1999.

D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, Cambridge, MA, 2009.

S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.

A. Lipshtat, H. B. Perets, N. Q. Balaban, and O. Biham. Modeling of negative autoregulated genetic networks in single cells. *Gene*, 347:265, 2005.

K. P. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, University of California, Berkeley, 2002.

B. Ng, A. Pfeffer, and R. Dearden. Continuous time particle filtering. In *Proc. of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI)*, 2005.

U. Nodelman, C. R. Shelton, and D. Koller. Continuous time Bayesian networks. In *Proc. Eighteenth Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 378–387, 2002.

U. Nodelman, C. R. Shelton, and D. Koller. Learning continuous time Bayesian networks. In *Proc. Nineteenth Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 451–458, 2003.

U. Nodelman, C. R. Shelton, and D. Koller. Expectation maximization and complex duration distributions for continuous time Bayesian networks. In *Proc. Twenty-first Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 421–430, 2005a.

U. Nodelman, C. R. Shelton, and D. Koller. Expectation propagation for continuous time Bayesian networks. In *Proc. Twenty-first Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 431–440, 2005b.

M. Opper and G. Sanguinetti. Variational inference for Markov jump processes. In *Advances in Neural Information Processing Systems 20*. MIT Press, 2007.

M. Opper and G. Sanguinetti. Learning combinatorial transcriptional dynamics from gene expression data. *Bioinformatics*, 26(13):1623–1629, 2010.

W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, New York, NY, USA, 3 edition, 2007.

S. Rajaram, T. Graepel, and R. Herbrich. Poisson-networks: A model for structured point processes. In *Proc. Tenth International Workshop on Artificial Intelligence and Statistics (AISTATS)*, January 2005.

A. Ruttor and M. Opper. Approximate parameter inference in a stochastic reaction-diffusion model. In *Proc. Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 9, pages 669–676, 2010.

G. Sanguinetti, A. Ruttor, M. Opper, and C. Archambeau. Switching regulatory models of cellular stress response. *Bioinformatics*, 25(10):1280–1286, 2009.

S. Saria, U. Nodelman, and D. Koller. Reasoning at the right time granularity. In *Proc. Twenty-third Conference on Uncertainty in Artificial Intelligence (UAI)*, 2007.

A. Simma, M. Goldszmidt, J. MacCormick, P. Barham, R. Black, R. Isaacs, and R. Mortier. Ct-nor: Representing and reasoning about events in continuous time. In *Proc. Twenty-fourth Conference on Uncertainty in Artificial Intelligence (UAI)*, 2008.

M. J. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.*, 1:1–305, 2008.

J. Yu and J. L. Thorne. Dependence among sites in RNA evolution. *Mol. Biol. Evol.*, 23:1525–37, 2006.

Chapter 5

Paper: Continuous-Time Belief Propagation

Tal El-Hay, Ido Cohn, Nir Friedman, and Raz Kupferman

In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, 2010

Continuous-Time Belief Propagation

Tal El-Hay[1] TALE@CS.HUJI.AC.IL
Ido Cohn[1] IDO_COHN@CS.HUJI.AC.IL
Nir Friedman[1] NIR@CS.HUJI.AC.IL
Raz Kupferman[2] RAZ@MATH.HUJI.AC.IL
[1]School of Computer Science and Engineering, [2]Institute of Mathematics, Hebrew University, Jerusalem 91904, Israel

Abstract

Many temporal processes can be naturally modeled as a stochastic system that evolves continuously over time. The representation language of *continuous-time Bayesian networks* allows to succinctly describe multi-component continuous-time stochastic processes. A crucial element in applications of such models is inference. Here we introduce a variational approximation scheme, which is a natural extension of Belief Propagation for continuous-time processes. In this scheme, we view messages as inhomogeneous Markov processes over individual components. This leads to a relatively simple procedure that allows to easily incorporate adaptive ordinary differential equation (ODE) solvers to perform individual steps. We provide the theoretical foundations for the approximation, and show how it performs on a range of networks. Our results demonstrate that our method is quite accurate on singly connected networks, and provides close approximations in more complex ones.

1. Introduction

The dynamics of many real-life processes are naturally modeled in terms of continuous-time stochastic processes, allowing for a wide range of time scales within the same process. Examples include biological sequence evolution(Felsenstein, 2004), computer systems (Xu & Shelton, 2008; Simma et al., 2008), and social networks (Fan & Shelton, 2009).

While the mathematical foundations of continuous-time stochastic processes are well understood (Chung, 1960), the study of efficient computer representations, inference, and learning of complex continuous-time processes is still in its early stages. *Continuous-time Bayesian networks* (CTBNs) (Nodelman et al., 2002) provide a sparse representation of complex multi-component processes by describing how the dynamics of an individual component depends on the state of its neighbors. A major challenge is translating the structure of a CTBN to computational gains in inference problems—answering queries about the process from partial observations.

As exact inference in a CTBN is exponential in the number of components, we have to resort to approximations. Broadly speaking, these fall into two main categories. The first category includes stochastic approximations (Fan & Shelton, 2008; El-Hay et al., 2008), which sample trajectories of the process. While these can be asymptotically exact, they can be computationally expensive and incur computational penalties when sampling rapidly evolving subprocesses. The second category of approximations includes variational methods. Nodelman et al. (2005) and Saria et al. (2007) developed an approach based on *expectation propagation* (Minka, 2001; Heskes & Zoeter, 2002), where the posterior distribution over a process is approximated by *piecewise homogeneous* factored processes. This involves an elaborate message passing scheme between the approximations for different components, and an adaptive procedure for determining how to segment each time interval. More recently, we introduced a mean-field approximation (Cohn et al., 2009), which uses factored *inhomogeneous* processes (Opper & Sanguinetti, 2007). This allowed us to build on the rich literature of adaptive ODE solvers. While the mean-field approximation provides a lower-bound on the likelihood, it suffers from the expected drawbacks when approximating highly coupled processes.

Here we introduce a variational approximation that combines insights from both previous approaches for variational inference in CTBNs. Our approximation is a natural extension of the successful Bethe approximation (Yedidia et al., 2005) to CTBNs. Alternatively, it can be viewed

Appearing in *Proceedings of the 27^{th} International Conference on Machine Learning*, Haifa, Israel, 2010. Copyright 2010 by the author(s)/owner(s).

applying the approach of Nodelman et al where the segment length diminishes to zero. Our approximation finds a collection of inhomogeneous processes over subsets of components, which are constrained to be locally consistent over single components. We show that this approximation is often accurate on tree-networks, and provides good approximations for more complex networks. Importantly, the approximation scheme is simple and allows to easily exploit the large suites of computational tools offered in the field of ODEs.

2. Continuous-Time Bayesian Networks

Consider a d-component Markov process $\boldsymbol{X}^{(t)} = (X_1^{(t)}, X_2^{(t)}, \ldots X_d^{(t)})$ with state space $S = S_1 \times S_2 \times \cdots \times S_d$. A notational convention: vectors are denoted by boldface symbols, e.g., $\boldsymbol{X}$, and matrices are denoted by blackboard style characters, e.g., $\mathbb{Q}$. The states in S are denoted by vectors of indexes, $\boldsymbol{x} = (x_1, \ldots, x_d)$. We use indexes $1 \leq i, j \leq d$ for enumerating components and $\boldsymbol{X}^{(t)}$ and $X_i^{(t)}$ to denote the random variable describing the state of the process and its i'th component at time t.

The dynamics of a *time-homogeneous continuous-time Markov process* are fully determined by the *Markov transition function*,

$$p_{\boldsymbol{x},\boldsymbol{y}}(t) = \Pr(\boldsymbol{X}^{(t+s)} = \boldsymbol{y} | \boldsymbol{X}^{(s)} = \boldsymbol{x}),$$

where time-homogeneity implies that the right-hand side does not depend on s. These dynamics are captured by a matrix $\mathbb{Q}$—the *rate matrix*, with non-negative off-diagonal entries $q_{\boldsymbol{x},\boldsymbol{y}}$ and diagonal entries $q_{\boldsymbol{x},\boldsymbol{x}} = -\sum_{\boldsymbol{y} \neq \boldsymbol{x}} q_{\boldsymbol{x},\boldsymbol{y}}$. The rate matrix is related to the transition function by

$$\left.\frac{d}{dt} p_{\boldsymbol{x},\boldsymbol{y}}(t)\right|_{t=0} = q_{\boldsymbol{x},\boldsymbol{y}}.$$

The probability of being in state $\boldsymbol{x}$ at time t satisfies the *master equation* (Chung, 1960)

$$\frac{d}{dt} Pr(\boldsymbol{X}^{(t)} = \boldsymbol{x}) = \sum_{\boldsymbol{y}} q_{\boldsymbol{y},\boldsymbol{x}} Pr(\boldsymbol{X}^{(t)} = \boldsymbol{y}).$$

A *continuous-time Bayesian network* is a structured multi-component continuous-time Markov process. It is defined by assigning each component i a set of components $\mathbf{Pa}_i \subseteq \{1, \ldots, d\} \setminus \{i\}$, which are its parents in the network (Nodelman et al., 2002). With each component i we then associate a family of rate matrices $\mathbb{Q}^{i|\mathbf{Pa}_i}_{\cdot|\boldsymbol{u}_i}$, with entries $q^{i|\mathbf{Pa}_i}_{x_i,y_i|\boldsymbol{u}_i}$, that describe the rates of change of the i'th component given the state $\boldsymbol{u}_i$ of the parents $\mathbf{Pa}_i$. The dynamics of $\boldsymbol{X}^{(t)}$ are defined by a rate matrix $\mathbb{Q}$ with entries $q_{\boldsymbol{x},\boldsymbol{y}}$ that combines the conditional rate matrices as follows:

$$q_{\boldsymbol{x},\boldsymbol{y}} = \begin{cases} q^{i|\mathbf{Pa}_i}_{x_i,y_i|\boldsymbol{u}_i} & \delta_{\boldsymbol{x},\boldsymbol{y}} = \{i\} \\ \sum_i q^{i|\mathbf{Pa}_i}_{x_i,x_i|\boldsymbol{u}_i} & \boldsymbol{x} = \boldsymbol{y} \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where $\delta_{\boldsymbol{x},\boldsymbol{y}} = \{i | x_i \neq y_i\}$. This definition implies that changes occur one component at a time.

Given a continuous-time Bayesian network, we would like to evaluate the likelihood of evidence, to compute the probability of various events given the evidence (e.g., that the state of the system at time t is $\boldsymbol{x}$), and to compute conditional expectations (e.g., the expected amount of time X_i was in state x_i). Direct computations of these quantities involve matrix exponentials of the rate matrix $\mathbb{Q}$, whose size is exponential in the number of components, making this approach infeasible beyond a modest number of components. We therefore have to resort to approximations.

3. A Variational Principle

Variational inference methods pose the inference task in terms of an optimization problem. The objective is to maximize a functional which lower-bounds the log probability of the evidence by introducing an auxiliary set of *variational parameters* (Wainwright & Jordan, 2008). Recently, we introduced a variational formulation of inference in continuous-time Markov processes. We start by reviewing the relevant results of Cohn et al.

For convenience we restrict our treatment to an interval $[0, T]$ with boundary evidence $\boldsymbol{X}^{(0)} = \boldsymbol{e}_0$ and $\boldsymbol{X}^{(T)} = \boldsymbol{e}_T$. The posterior distribution of a homogeneous Markov process given evidence $\boldsymbol{e} = \{\boldsymbol{e}_0, \boldsymbol{e}_T\}$ on the two boundaries is a *non-homogeneous Markov process*. Such a process can be represented using a *time varying rate matrix* $\mathbb{Q}(t)$ that describe the instantaneous transition rates. However, such a representation is unwieldy, since as t approaches T the transition rates from $\boldsymbol{x} \neq \boldsymbol{e}_T$ to $\boldsymbol{e}_T$ approach infinity.

To circumvent the problem of unbounded values near the boundaries, we introduced *marginal density sets* which represent the posterior process in terms of uni-variate and joint pairwise distributions. More formally, if $\Pr$ denotes the posterior distribution, its *marginal density set* is the following family of continuous functions:

$$\begin{aligned} \mu_{\boldsymbol{x}}(t) &= \Pr(\boldsymbol{X}^{(t)} = \boldsymbol{x}) \\ \gamma_{\boldsymbol{x},\boldsymbol{y}}(t) &= \lim_{h \downarrow 0} \frac{\Pr(\boldsymbol{X}^{(t)} = \boldsymbol{x}, \boldsymbol{X}^{(t+h)} = \boldsymbol{y})}{h}, \quad \boldsymbol{x} \neq \boldsymbol{y}. \end{aligned} \tag{2}$$

In addition to providing a bounded representation to the posterior, this representation allows to easily compute expected sufficient statistics using numerical integration:

$$\mathbf{E}\left[T_{\boldsymbol{x}}(t)\right] = \int_0^t \mu_{\boldsymbol{x}}(s)ds, \quad \mathbf{E}\left[M_{\boldsymbol{x},\boldsymbol{y}}(t)\right] = \int_0^t \gamma_{\boldsymbol{x},\boldsymbol{y}}(s)ds,$$

where $T_{\boldsymbol{x}}(t)$ is the residence time in state $\boldsymbol{x}$ in the interval $[0,t]$, and $M_{\boldsymbol{x},\boldsymbol{y}}(t)$ is the number of transitions from $\boldsymbol{x}$ to $\boldsymbol{y}$ in the same interval. Thus, this representation is analogous to sets of *mean parameters* that are employed in variational approximations over exponential families with a finite dimensional parametrization (Wainwright & Jordan, 2008; Koller & Friedman, 2009).

Families of functions μ, γ that satisfy (2) for some Pr, must satisfy self-consistent relations imposed by the master equation.

Definition 3.1 : (Cohn et al., 2009) A family $\eta = \{\mu_{\boldsymbol{x}}(t), \gamma_{\boldsymbol{x},\boldsymbol{y}}(t) : 0 \le t \le T\}$ of continuous functions is a *Markov-consistent density set* if the following constraints are fulfilled:

$$\begin{aligned}
\mu_{\boldsymbol{x}}(t) &\geq 0, \quad \sum_{\boldsymbol{x}} \mu_{\boldsymbol{x}}(0) = 1, \\
\gamma_{\boldsymbol{x},\boldsymbol{y}}(t) &\geq 0 \qquad \forall \boldsymbol{y} \neq \boldsymbol{x}, \\
\frac{d}{dt}\mu_{\boldsymbol{x}}(t) &= \sum_{\boldsymbol{y}\neq\boldsymbol{x}} \left(\gamma_{\boldsymbol{y},\boldsymbol{x}}(t) - \gamma_{\boldsymbol{x},\boldsymbol{y}}(t)\right).
\end{aligned}$$

and $\gamma_{\boldsymbol{x},\boldsymbol{y}}(t) = 0$ whenever $\mu_{\boldsymbol{x}}(t) = 0$. For convenience, we define $\gamma_{\boldsymbol{x},\boldsymbol{x}} = -\sum_{\boldsymbol{y}\neq\boldsymbol{x}} \gamma_{\boldsymbol{x},\boldsymbol{y}}$. ▮

The evidence at the boundaries impose additional constraints on potential posterior processes. Specifically, the representation η corresponding to the posterior distribution $P_{\mathbb{Q}}(\cdot|\boldsymbol{e}_0, \boldsymbol{e}_T)$ is in the set $\mathcal{M}_{\boldsymbol{e}}$ that contains Markov-consistent density sets $\{\mu_{\boldsymbol{x}}(t), \gamma_{\boldsymbol{x},\boldsymbol{y}}(t)\}$, that satisfy $\mu_{\boldsymbol{x}}(0) = \boldsymbol{1}_{\boldsymbol{x}=\boldsymbol{e}_0}$, $\mu_{\boldsymbol{x}}(T) = \boldsymbol{1}_{\boldsymbol{x}=\boldsymbol{e}_T}$ and $\gamma_{\boldsymbol{x}\boldsymbol{y}}(T) = 0$ for all $\boldsymbol{y} \neq \boldsymbol{e}_T$. In addition, since these sets are posteriors of a CTBN, they also change one component at a time, which implies that $\gamma_{\boldsymbol{x},\boldsymbol{y}}(t) = 0$ if $|\delta_{\boldsymbol{x},\boldsymbol{y}}| > 1$.

Using this representation, the variational formulation is reminiscent of similar formulations for discrete probabilistic models (Jordan et al., 1998).

Theorem 3.2: (Cohn et al., 2009) Let $\mathbb{Q}$ be a rate matrix and $\boldsymbol{e} = (\boldsymbol{e}_0, \boldsymbol{e}_T)$ be states of $\boldsymbol{X}$. Then

$$\ln P_{\mathbb{Q}}(\boldsymbol{e}_T|\boldsymbol{e}_0) = \max_{\eta\in\mathcal{M}_{\boldsymbol{e}}} \mathcal{F}(\eta;\mathbb{Q}),$$

where

$$\mathcal{F}(\eta;\mathbb{Q}) = \mathcal{E}(\eta;\mathbb{Q}) + \mathcal{H}(\eta),$$

is the *free energy functional* which is a sum of an *average energy functional*

$$\mathcal{E}(\eta;\mathbb{Q}) = \int_0^T \sum_{\boldsymbol{x}} \left[\mu_{\boldsymbol{x}}(t) q_{\boldsymbol{x},\boldsymbol{x}} + \sum_{\boldsymbol{y}\neq\boldsymbol{x}} \gamma_{\boldsymbol{x},\boldsymbol{y}}(t) \ln q_{\boldsymbol{x},\boldsymbol{y}}\right] dt,$$

and an *entropy functional*

$$\mathcal{H}(\eta) = \int_0^T \sum_{\boldsymbol{x}} \sum_{\boldsymbol{y}\neq\boldsymbol{x}} \gamma_{\boldsymbol{x},\boldsymbol{y}}(t)[1 + \ln\mu_{\boldsymbol{x}}(t) - \ln\gamma_{\boldsymbol{x},\boldsymbol{y}}(t)]dt \ .$$

To illustrate this principle, we can examine how to derive an exact inference procedure. We can find the optimum of $\mathcal{F}(\eta;\mathbb{Q})$ by introducing Lagrange multipliers that enforce the consistency constraint, and then find the stationary point of the corresponding Lagrangian. Since we are dealing with a continuous-time formula, we need to use the Euler-Lagrange method . As we show, the maximum satisfies a system of differential equations:

$$\begin{aligned}
\frac{d}{dt}\rho_{\boldsymbol{x}} &= -\sum_{\boldsymbol{y}} q_{\boldsymbol{x},\boldsymbol{y}}\rho_{\boldsymbol{y}} & \rho_{\boldsymbol{x}}(T) &= \boldsymbol{1}_{\boldsymbol{x}=\boldsymbol{e}_T} \\
\frac{d}{dt}\mu_{\boldsymbol{x}} &= \sum_{\boldsymbol{y}\neq\boldsymbol{x}} (\gamma_{\boldsymbol{y},\boldsymbol{x}} - \gamma_{\boldsymbol{x},\boldsymbol{y}}), & \mu_{\boldsymbol{x}}(0) &= \boldsymbol{1}_{\boldsymbol{x}=\boldsymbol{e}_0} \\
\gamma_{\boldsymbol{x},\boldsymbol{y}} &= \mu_{\boldsymbol{x}} q_{\boldsymbol{x},\boldsymbol{y}} \frac{\rho_{\boldsymbol{y}}}{\rho_{\boldsymbol{x}}}, & \boldsymbol{y} &\neq \boldsymbol{x}, \rho_{\boldsymbol{x}} \neq 0,
\end{aligned} \tag{3}$$

where we omit the (t) argument for clarity. The auxiliary functions $\rho_{\boldsymbol{x}}(t)$ are Lagrange multipliers.

These equations have a simple intuitive solution that involves backward integration of $\rho_{\boldsymbol{x}}(t)$, as we have a boundary condition at time T and $\rho_{\boldsymbol{x}}(t)$ does not depend on $\mu_{\boldsymbol{x}}(t)$. This integration results in

$$\rho_{\boldsymbol{x}}(t) = \Pr(\boldsymbol{e}_T|\boldsymbol{X}^{(t)} = \boldsymbol{x})$$

Once we solve for $\rho_{\boldsymbol{x}}(t)$, we can forward integrate $\mu_{\boldsymbol{x}}(t)$ from the boundary conditions at 0 to get the solution for $\mu_{\boldsymbol{x}}$ and $\gamma_{\boldsymbol{x},\boldsymbol{y}}$. This analysis suggests that this system of ODEs is similar to forward-backward propagation, except that unlike classical forward propagation, here the forward propagation takes into account the backward messages to directly compute the posterior. Note that applying this exact solution to a multi-component process results in an exponential (in d) number of coupled differential equations.

4. Continuous-Time Expectation Propagation

Approximate variational inference procedures are derived by posing an optimization problem that is an approximate version of the original one. Different approximations differ in terms of whether they approximate the objectives, limit or relax the allowed set of solutions, or combine several such approaches. Here, we follow a strategy which is based on the approach of *expectation propagation*, in which the set of admissible solutions is extended to ones that are consistent only on the expectations of statistics of interest, and in addition, use an approximate functional.

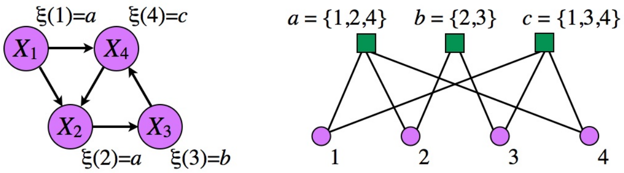


Figure 1. A CTBN and a corresponding factor graph.

4.1. Approximate Optimization Problem

To represent potential solutions, we follow methods used in recent approximate inference procedures that use factor graph representations (Yedidia et al., 2005; Koller & Friedman, 2009). Specifically, we keep only marginal density sets over smaller clusters of components.

We start with definitions and notations. A *factor graph* is an undirected bipartite graph. One layer in the graph consists of *component nodes* that are labeled by component indexes. The second layer consists of *clusters nodes* $\mathcal{A}$, where each cluster $\alpha \in \mathcal{A}$, is a subset of $\{1, \ldots, d\}$. The edges in the graph are between a component node i to a cluster node α if and only if $i \in \alpha$. Thus, the neighbors of α are $N(\alpha) = \{i : i \in \alpha\}$ and the neighbors of i are $N(i) = \{\alpha : i \in \alpha\}$.

A factor graph is *family preserving*, with respect to a given CTBN, if there exists an assignment function $\xi(i)$ that maps components to clusters, such that for every i, we have that $\{i\} \cup \mathbf{Pa}_i \subseteq \xi(i)$. We denote by $A(\alpha)$ the set of components i for which $\xi(i) = \alpha$. From now on, we assume that we deal only with family preserving factor graphs.

Example 4.1: Figure 1 shows a simple CTBN and a corresponding factor graph. In this specific factor graph, $\mathcal{A}(a) = \{1, 2\}$, $\mathcal{A}(b) = \{3\}$ and $\mathcal{A}(c) = \{4\}$. ∎

Given a factor graph, we use its structure to define an approximation for a distribution. Instead of describing the distribution over all the components, we use a family of density sets $\tilde{\boldsymbol{\eta}} = \{\eta^i : i = 1, \ldots, d\} \cup \{\eta^\alpha : \alpha \in \mathcal{A}\}$. A family of marginal density sets can be inconsistent. We do not require full consistency, but only consistency between neighboring nodes in the following sense.

Definition 4.2: A family of density sets $\tilde{\boldsymbol{\eta}}$ is said to be *locally consistent* if for all $\alpha \in \mathcal{A}$ and all $i \in N(\alpha)$ we have $\mu^i = \mu^\alpha|_i$ where

$$\left(\mu^\alpha|_i\right)_{x_i} = \sum_{\boldsymbol{x}_{\alpha\setminus i}} \mu^\alpha_{[\boldsymbol{x}_{\alpha\setminus i}, x_i]} \tag{4}$$

and $[\boldsymbol{x}_{\alpha\setminus i}, x_i]$ is the assignment to $\boldsymbol{x}_\alpha$ composed from $\boldsymbol{x}_{\alpha\setminus i}$ and x_i. Likewise, $\gamma^i = \gamma^\alpha|_i$ where

$$\left(\gamma^\alpha|_i\right)_{x_i, y_i} = \sum_{\boldsymbol{x}_{\alpha\setminus i}} \gamma^\alpha_{[\boldsymbol{x}_{\alpha\setminus i}, x_i], [\boldsymbol{x}_{\alpha\setminus i} y_i]}. \tag{5}$$

Let $\tilde{\mathcal{M}}_{\boldsymbol{e}}$ be the set of locally consistent densities that correspond to evidence $\boldsymbol{e}$ ∎

The local consistency of η^α and η^i does not imply that the distribution $\Pr_{\eta^i}(X_i)$ is equal to the marginal distribution $\Pr_{\eta^\alpha}(X_i)$, as marginalization of a Markov process is not necessarily a Markov process. Rather, $\Pr_{\eta^i}$ is the projection of $\Pr_{\eta^\alpha}(X_i)$ to a Markov process with the matching expectations of $\mathbf{E}\left[T_{x_i}(t)\right]$ and $\mathbf{E}\left[M_{x_i,y_i}(t)\right]$ (Koller & Friedman, 2009).

Such locally consistent sets allow us to construct a tractable approximation to the variational optimization problem by introducing the *continuous-time Bethe functional*

$$\tilde{\mathcal{F}}(\tilde{\boldsymbol{\eta}}; \mathbb{Q}) = \sum_i \mathcal{E}_i(\eta^{\xi(i)}; \mathbb{Q}^{i|\mathbf{Pa}_i}) + \sum_\alpha \mathcal{H}(\eta^\alpha) - \sum_i c_i \mathcal{H}(\eta^i)$$

where, for $\alpha = \xi(i)$,

$$\mathcal{E}_i(\eta^\alpha; \mathbb{Q}^{i|\mathbf{Pa}_i}) = \int_0^T \sum_{\boldsymbol{x}_\alpha} \left[\mu^\alpha_{\boldsymbol{x}_\alpha}(t) q^{i|\mathbf{Pa}_i}_{x_i,x_i|\boldsymbol{u}_i} + \sum_{\boldsymbol{y}\neq\boldsymbol{x}} \gamma^\alpha_{\boldsymbol{x},\boldsymbol{y}}(t) \ln q^{i|\mathbf{Pa}_i}_{x_i,y_i|\boldsymbol{u}_i}\right] dt,$$

and $c_i = N(i) - 1$ ensure that the total weight of sets containing component i sums up to 1. This functional is analogous to the well-known Bethe approximation for discrete models (Yedidia et al., 2005).

Combining the two approximations the approximate optimization problem becomes:

$$\max_{\tilde{\boldsymbol{\eta}} \in \tilde{\mathcal{M}}_{\boldsymbol{e}}} \tilde{\mathcal{F}}(\tilde{\boldsymbol{\eta}}; \mathbb{Q}) \tag{6}$$

Once the optimal parameters are found, we can use the relevant marginal density set to answer queries.

4.2. Stationary Point Characterization

To characterize the stationary points of the approximate optimization problem (6) we use again the Euler-Lagrange method, where we introduce Lagrange multiplier functions to enforce the cluster-wise constraints, $\frac{d}{dt}\mu^\alpha_{\boldsymbol{x}_\alpha} = \sum_{\boldsymbol{y}\neq\boldsymbol{x}}(\gamma^\alpha_{\boldsymbol{y}_\alpha,\boldsymbol{x}_\alpha} - \gamma^\alpha_{\boldsymbol{x}_\alpha,\boldsymbol{y}_\alpha})$ as well as the local consistency constraints defined in equations (4) and (5). Differentiating the Lagrangian, equating the derivatives to zero, and performing some algebra, which we omit for the lack of space, we obtain fixed-point equations that consist of the initial constraints and two classes of coupled equations.

The first class consists of equations similar to (3), which refer to the dynamics within each cluster. To simplify the presentation, we introduce some definitions.

Definition 4.3: Assume we are given a time-varying matrix function $\mathbb{G}(t)$, and boundary conditions $\boldsymbol{x}_0$ and $\boldsymbol{x}_T$. Define the operator $\eta = \mathcal{R}(\mathbb{G}, \boldsymbol{x}_0, \boldsymbol{x}_T)$ to return $\eta = (\mu, \gamma)$, the unique solution of the following ODEs

$$\begin{aligned} \frac{d}{dt}\rho_{\boldsymbol{x}} &= -\sum_{\boldsymbol{y}} g_{\boldsymbol{x},\boldsymbol{y}}\rho_{\boldsymbol{y}}, & \rho_{\boldsymbol{x}}(T) &= \boldsymbol{1}_{\boldsymbol{x}=\boldsymbol{x}_T} \\ \frac{d}{dt}\mu_{\boldsymbol{x}} &= \sum_{\boldsymbol{y}\neq\boldsymbol{x}} (\gamma_{\boldsymbol{y},\boldsymbol{x}} - \gamma_{\boldsymbol{x},\boldsymbol{y}}), & \mu_{\boldsymbol{x}}(0) &= \boldsymbol{1}_{\boldsymbol{x}=\boldsymbol{x}_0} \\ \gamma_{\boldsymbol{x},\boldsymbol{y}} &= \mu_{\boldsymbol{x}} g_{\boldsymbol{x},\boldsymbol{y}} \frac{\rho_{\boldsymbol{y}}}{\rho_{\boldsymbol{x}}}, & \rho_{\boldsymbol{x}} &\neq 0, \boldsymbol{y} \neq \boldsymbol{x}. \end{aligned}$$

∎

Note that this set of equations is identical to (3), but replaces the constant rate matrix $\mathbb{Q}$ by a time varying matrix function $\mathbb{G}(t)$. Using this terminology, the first part of the fixed-point equations is

$$\eta^{\alpha} = \mathcal{R}(\mathbb{G}^{\alpha}, \boldsymbol{e}_0|_{\alpha}, \boldsymbol{e}_T|_{\alpha}), \tag{7}$$

where $\mathbb{G}^{\alpha}(t)$ is the time-dependent matrix with entries

$$g^{\alpha}_{\boldsymbol{x}_\alpha,\boldsymbol{y}_\alpha} = \tag{8}$$
$$\begin{cases} (q^{i|\mathbf{Pa}_i}_{x_i y_i|\boldsymbol{u}_i})^{\boldsymbol{1}_{i\in A(\alpha)}} \cdot n^{i\to\alpha}_{x_i,y_i} & \delta_{\boldsymbol{x}_\alpha,\boldsymbol{y}_\alpha} = \{i\} \\ \sum_{i\in N(\alpha)} \left(\boldsymbol{1}_{i\in A(\alpha)} q^{i|\mathbf{Pa}_i}_{x_i x_i|\boldsymbol{u}_i} + n^{i\to\alpha}_{x_i,x_i} \right) & \boldsymbol{x}_\alpha = \boldsymbol{y}_\alpha \\ 0 & \text{otherwise}, \end{cases}$$

and $n^{i,\alpha}$ are time-dependent functions that originate from the Lagrange multipliers that enforce local consistency constraints,

$$\begin{aligned} \prod_{\alpha\in N(i)} n^{i\to\alpha}_{x_i,y_i} &= \left(\frac{\gamma^i_{x_i y_i}}{\mu^i_{x_i}}\right)^{c_i}, \quad x_i \neq y_i \\ \sum_{\alpha\in N(i)} n^{i\to\alpha}_{x_i,x_i} &= c_i \frac{\gamma^i_{x_i x_i}}{\mu^i_{x_i}}. \end{aligned} \tag{9}$$

These equations together with, (4) and (5) form the second set of equations that couple different clusters.

Equation (7) suggests that the matrix $\mathbb{G}^{\alpha}$ plays the role of a rate matrix. Unlike $\mathbb{Q}$, $\mathbb{G}^{\alpha}$ is not guaranteed to be a rate matrix as its rows do not necessarily sum up to zero. Nonetheless, even though it is not a rate matrix, this system of equations has a unique solution that can be found using a backward-forward integration. Thus, since the matrix function $\mathbb{G}^{\alpha}$ corresponds to a unique density set, we say that $\mathbb{G}^{\alpha}$ is an *unnormalized parametrization* of the process $P_{\eta^{\alpha}}$.

At this point, it is tempting to proceed to construct a message passing algorithm based on this fixed point characterization in analogous manner to the developments of Yedidia et al. (2005) . However, we are faced with a problem. Note that $\lim_{t\to T} \frac{\gamma_{x_i e_i}}{\mu_{x_i}} = \infty$. Therefore, according to Equation (9), when t approaches T, there exists some $\alpha \in N(i)$ for which $n^{i,\alpha}_{x_i e_{T,i}}(t)$ approaches ∞ as $t \to T$. This implies that a simple-minded message passing procedure is susceptible to unbounded values and numerical difficulties.

4.3. Gauge Transformation

To overcome these numerical difficulties, we now derive an alternative characterization, which does not suffer from unbounded values. We start with a basic result.

Proposition 4.4: *Let $\mathbb{G}$ be a unnormalized rate matrix function, and let $\omega_{\boldsymbol{x}}(t)$ be a smooth positive vector-valued function, where $\omega_{\boldsymbol{x}}(t) > 0$ in $[0, T)$. Let $\mathbb{G}^{\omega}$ to be the matrix function with*

$$g^{\omega}_{\boldsymbol{x}\boldsymbol{y}} = \begin{cases} g_{\boldsymbol{x}\boldsymbol{y}} \cdot \frac{\omega_{\boldsymbol{x}}}{\omega_{\boldsymbol{y}}} & \boldsymbol{y} \neq \boldsymbol{x} \\ g_{\boldsymbol{x}\boldsymbol{x}} - \frac{d}{dt}\log\omega_{\boldsymbol{x}} & \boldsymbol{y} = \boldsymbol{x}. \end{cases} \tag{10}$$

Then, $\mathcal{R}(\mathbb{G}, \boldsymbol{x}_0, \boldsymbol{x}_T) = \mathcal{R}(\mathbb{G}^{\omega}, \boldsymbol{x}_0, \boldsymbol{x}_T)$.

Proof sketch: Let ρ, η satisfy the system of equations of Def. 4.3 with $\mathbb{G}$. Define $\rho^{\omega} = \rho \cdot \omega$, and show that ρ^{ω}, η satisfy the same system of equations with $\mathbb{G}^{\omega}$. ∎

This result characterizes transformations of (8–9) that do not change the fixed point solutions for cluster density sets. We seek transformations that reweigh the functions $n^{i,\alpha}$ so that they remain bounded using the following result.

Proposition 4.5: *Assume $\mathbb{G}$ is a unnormalized rate matrix function such that $g_{\boldsymbol{x},\boldsymbol{y}}(t) \neq 0$ for all $\boldsymbol{x}, \boldsymbol{y}$, $g_{\boldsymbol{x},\boldsymbol{y}}(t)$ is continuously differentiable in $[0, T]$, and $\eta = \mathcal{R}(\mathbb{G}, \boldsymbol{x}_0, \boldsymbol{x}_T)$. If $\omega(t)$ is a family of smooth functions satisfying $\omega_{\boldsymbol{x}}(T) = \boldsymbol{1}_{\boldsymbol{x}=\boldsymbol{x}_T}$ and $\frac{d}{dt}\omega_{\boldsymbol{x}}(T) < 0$ for $\boldsymbol{x} \neq \boldsymbol{x}_T$, then*

$$\lim_{t\to T} \frac{\gamma_{\boldsymbol{x},\boldsymbol{y}}(t)}{\mu_{\boldsymbol{x}}(t)} \frac{\omega_{\boldsymbol{x}}(t)}{\omega_{\boldsymbol{y}}(t)} < \infty, \qquad \forall \boldsymbol{x} \neq \boldsymbol{y}$$

and

$$\lim_{t\to T} \left(\frac{\gamma_{\boldsymbol{x},\boldsymbol{x}}(t)}{\mu_{\boldsymbol{x}}(t)} - \frac{d}{dt}\log\omega_{\boldsymbol{x}}(t) \right) < \infty, \qquad \forall \boldsymbol{x}.$$

Example 4.6: One function that satisfies the conditions of Proposition 4.5 is $\omega_{\boldsymbol{x}}(t) = 1 - t/T, \forall \boldsymbol{x} \neq \boldsymbol{e}_T$ and $\omega_{\boldsymbol{e}_T}(t) = 1$. ∎

Using this result, we introduce weight functions $\omega^i_{x_i}$ (as above) and define $\omega^{\alpha}_{\boldsymbol{x}_\alpha} = \prod_{i\in N(\alpha)} (\omega^i_{x_i})^{c_i/(c_i+1)}$. Using these weight functions, we define $m^{i\to\alpha}_{x_i,y_i} = n^{i\to\alpha}_{x_i,y_i} (\frac{\omega^i_{x_i}}{\omega^i_{y_i}})^{c_i/(c_i+1)}$ and $m^{i\to\alpha}_{x_i,x_i} = n^{i\to\alpha}_{x_i,x_i} - \frac{c_i}{c_i+1}\frac{d}{dt}\log\omega^i_{x_i}$. Now if we define the time-dependent matrix $\tilde{\mathbb{G}}^{\alpha}$ with entries

$$\tilde{g}^{\alpha}_{\boldsymbol{x}_\alpha,\boldsymbol{y}_\alpha} = \tag{11}$$
$$\begin{cases} (q^{i|\mathbf{Pa}_i}_{x_i y_i|\boldsymbol{u}_i})^{\boldsymbol{1}_{i\in A(\alpha)}} \cdot m^{i\to\alpha}_{x_i,y_i} & \delta_{\boldsymbol{x}_\alpha,\boldsymbol{y}_\alpha} = \{i\} \\ \sum_{i\in N(\alpha)} \left(\boldsymbol{1}_{i\in A(\alpha)} q^{i|\mathbf{Pa}_i}_{x_i x_i|\boldsymbol{u}_i} + m^{i\to\alpha}_{x_i,x_i} \right) & \boldsymbol{x}_\alpha = \boldsymbol{y}_\alpha \\ 0 & \text{otherwise}, \end{cases}$$

Algorithm 1 Continuous-Time Belief Propagation

```
Initialize messages: for all $\alpha$ and all $i \in N(\alpha)$
    Choose $\eta^\alpha \in \mathcal{M}^\alpha_{\boldsymbol{e}}$
    Compute $\eta^{\alpha\to i}$ using (14)
    Set $m^{i\to\alpha}_{x_i,y_i} = 1 \; \forall x_i \neq y_i$, $m^{i\to\alpha}_{x_i,x_i} = 0$
repeat
    Choose a cluster $\alpha$:
    1. $\forall i \in N(\alpha)$, set $m^{i\to\alpha}_{x_i,y_i}$ using (15)
    2. Update $\tilde{\mathbb{G}}^\alpha$ using (11)
    3. Compute $\eta^\alpha$ from $\tilde{\mathbb{G}}^\alpha$ using (12)
    4. $\forall i \in N(\alpha)$ compute $\eta^{\alpha\to i}$ using (14)
until convergence
```

then $\tilde{\mathbb{G}}^\alpha = (\mathbb{G}^\alpha)^{\omega^\alpha}$. By Proposition 4.4,

$$\eta^\alpha = \mathcal{R}(\tilde{\mathbb{G}}^\alpha, \boldsymbol{e}_0|_\alpha, \boldsymbol{e}_T|_\alpha). \tag{12}$$

Plugging the definition of $m^{i\to\alpha}_{x_i,y_i}$ and $m^{i\to\alpha}_{x_i,x_i}$ into (9) we get

$$\begin{aligned}\prod_{\alpha\in N(i)} m^{i\to\alpha}_{x_i,y_i} &= \left(\frac{\gamma^i_{x_iy_i}}{\mu^i_{x_i}}\frac{\omega^i_{x_i}}{\omega^i_{y_i}}\right)^{c_i}, \quad x_i \neq y_i \\ \sum_{\alpha\in N(i)} m^{i\to\alpha}_{x_i,x_i} &= c_i\left(\frac{\gamma^i_{x_ix_i}}{\mu^i_{x_i}} - \frac{d}{dt}\log\omega^i_{x_i}\right).\end{aligned} \tag{13}$$

If the preconditions of Proposition 4.5 are satisfied, the terms in (13) are bounded. Together (11)–(13) provide an alternative characterization of the fixed point(s) of the optimization problem.

4.4. Message Passing Scheme

We now use the above characterization as justification for a message passing scheme, that if converged, will satisfy the fixed point equations. While (11) and (12) are readily transformed into assignments, (13) poses a challenge.

We start by noting that (13) contains the terms $\mu^i_{x_i}$ and $\gamma^i_{x_i,y_i}$. We can get these terms from η^α for *any* $\alpha \in N(i)$. Thus, for $\alpha \in N(i)$, we define

$$\mu^{\alpha\to i} = \mu^\alpha|_i \qquad \gamma^{\alpha\to i} = \gamma^\alpha|_i \tag{14}$$

We view these as the messages from cluster α to the component i. At convergence, $\mu^{\alpha\to i} = \mu^{\beta\to i}$ for $\alpha, \beta \in N(i)$, but this is not true before convergence.

Next, we rewrite (13) as an assignment

$$m^{i\to\alpha}_{x_i,y_i} = \begin{cases} \displaystyle\prod_{\substack{\beta\in N(i)\\ \beta\neq\alpha}} \frac{1}{m^{i\to\beta}_{x_i,y_i}} \frac{\gamma^{\beta\to i}_{x_iy_i}}{\mu^{\beta\to i}_{x_i}} \frac{\omega^i_{x_i}}{\omega^i_{y_i}} & x_i \neq y_i \\ \displaystyle\sum_{\substack{\beta\in N(i)\\ \beta\neq\alpha}} \left(\frac{\gamma^{\beta\to i}_{x_ix_i}}{\mu^{\beta\to i}_{x_i}} - \frac{d}{dt}\log\omega^i_{x_i} - m^{i\to\beta}_{x_i,x_i}\right) & \text{else,} \end{cases} \tag{15}$$

where we write $\frac{\gamma^i_{x_iy_i}}{\mu^i_{x_i}} = \frac{\gamma^{\beta\to i}_{x_iy_i}}{\mu^{\beta\to i}_{x_i}}$ once for each β.

The algorithm is summarized in Algorithm 1. The implementation of these steps involve a few details. We start with the initialization of messages. The only free parameter is the initial values of η^α. To ensure that these initial choices are in $\mathcal{M}^\alpha_{\boldsymbol{e}}$, we choose initial rates, and perform computations to get a valid posterior for the clusters. Another degree of freedom is the order of cluster updates. We use a randomized strategy, choosing a cluster at random, and if one of its neighbors was updated since it was last chosen, we update it. The computation in Step 3, involves reverse integration followed by forward integration (as explained in Section 3) . We gain efficiency by using adaptive numerical integration procedures. Specifically, we use the Runge-Kutta-Fehlberg (4,5) method. This method chooses temporal evaluation points on the fly, and returns values at these points. The computations of Step 2 is done on demand only at the evaluation points. To allow efficient interpolation, we use a piecewise linear approximation of η whose boundary points are determined by the evaluation points that are chosen by the adaptive integrator. Finally, as might be expected, we do not have convergence guarantees. However, if the algorithm converges, the fixed point equations are satisfied, hence giving a stationary point (hopefully a local maximum) of problem (6).

5. Experiments

We tested our method on three representative network structures: a directed tree, a directed toroid, and a bidirectional ring (Fig. 2). The tree network does not have any cycles. The toroid network has cycles, but these are fairly long, whereas the bidirectional ring has multiple short cycles. All networks are parametrized as *dynamic Ising models*, in which neighboring components prefer to be in the same state. Specifically, we use the conditional rates

$$q^{i|\mathbf{Pa}_i}_{x_i,y_i|\boldsymbol{u}_i} = \tau\left(1 + \exp\left(-2y_i\beta\sum\nolimits_{j\in\mathbf{Pa}_i} x_j\right)\right)^{-1}$$

where $x_j \in \{-1, 1\}$, β is a *coupling parameter* and τ is a *rate parameter*. Small values of β correspond to weak coupling, whereas τ determines how fast components tend to switch states. For each experiment we set evidence at times 0 and 1 (Fig. 2, left panel).

We compare the Bethe approximation to exact inference and mean-field (Cohn et al., 2009). We start by comparing the value of sufficient statistics (residence time and number of transitions of each component for each state of its parents) computed by each method. For example, for a particular choice of β and τ, (Fig. 2 middle columns) we can see that the Bethe approximation is virtually exact on the tree model and the toroid, but has some bias on the bidirectional ring model. These scatter plots also shed light

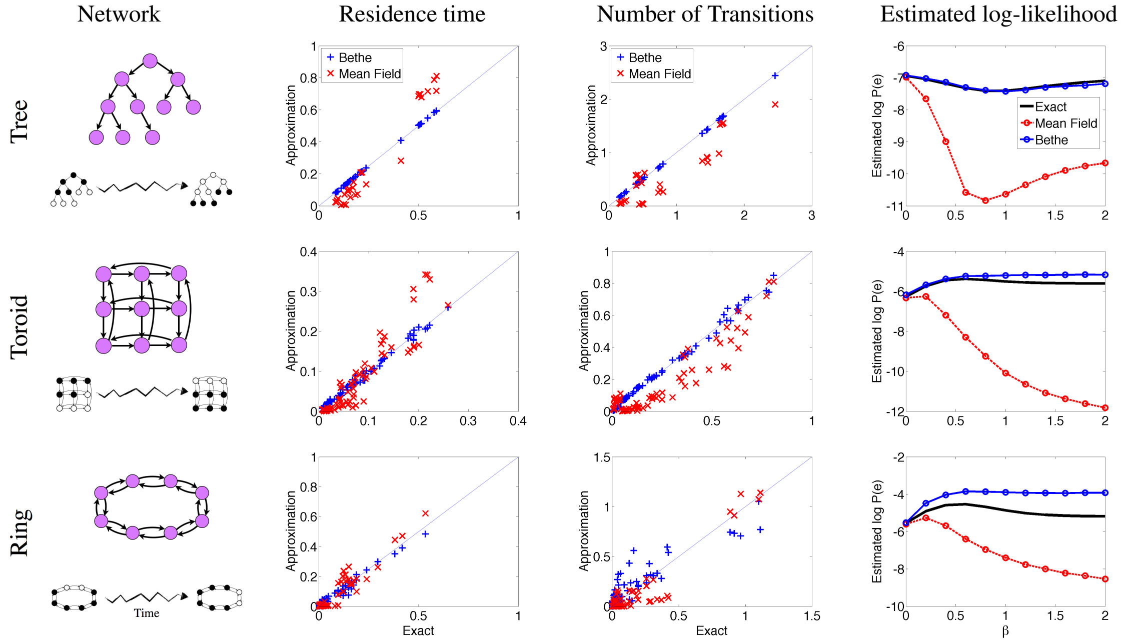


Figure 2. Simulation results for a tree network (top row), a toroid network (middle), and a bidirectional chain (bottom). **Left** network structure and the evidence at start and end points; black is $+1$ and white is -1. **Middle-left**: scatter plot of expected conditional residence times for networks with $\beta = 1$, $\tau = 8$. Each point corresponds to a single statistic, the x-axis is the exact value and the y-axis is the approximate value. **Middle-right**: same for expected conditional transition numbers. **Right**: exact and approximations of log-likelihood as function of β, the strength of coupling between components ($\tau = 8$).

on the nature of the difference between the two methods. Specifically, in the most likely scenario, two components switch from -1 to 1 near the beginning and the other two switch from 1 to -1 near the end, and so through most of the interval all the components are in the same state. The mean-field algorithm gives a uni-modal solution, focusing on the most likely scenario, resulting in zero residence time for the less likely states. These states are represented by the points close on the x-axis. The Bethe representation on the other hand can capture multiple scenarios.

Another aspect of the approximation is the estimation of the likelihood. In Fig. 2 (right column) we compare these estimations as function of β, the problem hardness. Again, we see that the Bethe approximation is essentially exact on the tree network, and provides close approximations in the two other networks. When we push β and τ to extreme values we do see inaccuracies even in the tree network, showing that the algorithm is an approximation.

While the ODE solvers used here allow adaptive integration error control, we do not have an a-priory control on the propagation of this error. To test this effect on overall accuracy, we repeated these experiments using standard grid refinement. Specifically, we computed integrals using uniformly spaced evaluation points and systematically halving integration intervals until no changes in the output were observed. Final results of these tests were practically the same as those obtained using adaptive integration.

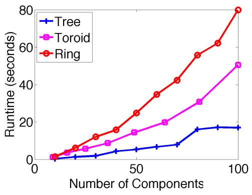


Figure 3. Run time vs. the number components in the three networks types ($\beta = 1$, $\tau = 8$).

Next, we examine how the algorithm scales with the number of components in the networks. In all three networks we see that the magnitude of relative error is essentially independent of the number of components (not shown). Fig. 3 shows that the run time scales linearly with the number of components. In harder networks the algorithm requires more iterations leading to slower convergence.

6. Discussion

Here, we introduce a message passing scheme that provides a variational approximation for CTBNs. This scheme is derived from an approximate free energy functional, based on the principles of expectation-propagation, where we require the posteriors on clusters to be locally consistent in terms of the Markovian projections on individual components. We show that stationary points of this optimization problem are the fixed points of a message passing algorithm, whose structure closely mirrors Belief Propagation.

In contrast to Belief Propagation on discrete (atemporal) networks, our algorithm is *not* guaranteed to be exact on tree CTBNs. The source of inaccuracy is the projection of the marginal distributions over components onto Markov processes. While this projection looses information, our empirical results suggest that this approximation is relatively accurate.

The works that are closest to ours are those of Nodelman et al. (2005) and Saria et al. (2007) which are also derived from an expectation-propagation energy functional. The main difference between the two methods is the structure of the approximation. Nodelman et al use a piecewise homogeneous representation, allowing them to represent the rate in each homogeneous segment by a constant (conditional) rate matrix. This, however, requires introducing machinery for deciding how to segment each component. As Saria et al show, this choice can have dramatic impact on the quality of the approximation and the running time. In contrast, our approach uses a (continuously) inhomogeneous representation, which is essentially the limit when segment sizes tend to zero. Surprisingly, rather than making the problem more complex, this choice simplifies the mathematics and also the implementation. In particular, our solution decouples the probabilistic issues (dependencies between components) and numerical issues (adaptive integration) and allows us to build on well-understood methods from numerical integration for efficient and adaptive selection of the number and placement of discretization points.

Our results show how a careful choice of representations and operations over them can narrow the gap between inference methods in discrete and continuous-time graphical models. Our constructions can be naturally generalized to capture more complex dependencies using methods based on Generalized Belief Propagation (Yedidia et al., 2005).

Acknowledgments

We thank Ofer Meshi and the anonymous reviewers for helpful remarks on previous versions of the manuscript. This research was supported in part by a grant from the Israel Science Foundation. Tal El-Hay is supported by the Eshkol fellowship from the Israeli Ministry of Science.

References

Chung, K.L. *Markov chains with stationary transition probabilities*. 1960.

Cohn, I., El-Hay, T., Friedman, N., and Kupferman, R. Mean field variational approximation for continuous-time Bayesian networks. UAI, 2009.

El-Hay, T., Friedman, N., and Kupferman, R. Gibbs sampling in factorized continuous-time markov processes. UAI, 2008.

Fan, Y. and Shelton, C. R. Learning continuous-time social network dynamics. UAI, 2009.

Fan, Y. and Shelton, C.R. Sampling for approximate inference in continuous time Bayesian networks. In *AI & Math*, 2008.

Felsenstein, J. *Inferring Phylogenies*. Sinauer, 2004.

Heskes, T. and Zoeter, O. Expectation propagation for approximate inference in dynamic Bayesian networks. UAI, 2002.

Jordan, M.I., Ghahramani, Z., Jaakkola, T.S., and Saul, L. K. An introduction to variational approximations methods for graphical models. In *Learning in Graphics Models*. 1998.

Koller, D. and Friedman, N. *Probabilistic Graphical Models: Principles and Techniques*. 2009.

Minka, T. P. Expectation propagation for approximate Bayesian inference. UAI, 2001.

Nodelman, U., Shelton, C. R., and Koller, D. Continuous time Bayesian networks. UAI, 2002.

Nodelman, U., Shelton, C.R., and Koller, D. Expectation propagation for continuous time Bayesian networks. UAI, 2005.

Opper, M. and Sanguinetti, G. Variational inference for Markov jump processes. NIPS, 2007.

Saria, S., Nodelman, U., and Koller, D. Reasoning at the right time granularity. UAI, 2007.

Simma, A., Goldszmidt, M., MacCormick, M., Barham, M., Black, M., Isaacs, M., and Mortier, R. CT-NOR: Representing and reasoning about events in continuous time. UAI, 2008.

Wainwright, M. J. and Jordan, M. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.*, 1:1–305, 2008.

Xu, J. and Shelton, C. R. Continuous time Bayesian networks for host level network intrusion detection. ECML/PKDD, 2008.

Yedidia, J.S., Freeman, W.T., and Weiss, Y. Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Trans. Info. Theory*, 51:2282–2312, 2005.

Chapter 6

Discussion

In this dissertation I have presented models and tools for probabilistic reasoning about structured stochastic systems in continuous-time. This framework is natural for many dynamic systems, potentially leading both to succinct and intuitive models as well as to efficient and relatively simple algorithms. The first part of the dissertation introduces the modeling language of continuous-time Markov networks. This language allows us to learn a compact representation of the stationary distribution of a process even when the sampled process has not reached equilibrium. As this task requires inferring expected sufficient statistics, we provide a transformation from CTMNs to CTBNs, allowing us to exploit CTBN inference algorithms. This leads us to deal with the inference problem on CTBN's, which is the main computational bottleneck in learning procedures. We begun by introducing a Gibbs sampling algorithm providing asymptotically unbiased estimates. Then, using a variational approach we derive to algorithms that are biased but are generally more efficient than sampling.

6.1 Modeling and Learning the Stationary Distribution

Continuous-time Markov networks describe processes that are composed of two major forces: a local proposal stochastic process acting independently on each component; and a global probabilistic selection function parameterized by a distribution over the states of the system. We showed that the distribution that determines selection is also the stationary distribution of the process.

In general, the stationary distribution reflects the propensity of the system to be in different states. Modeling this distribution as a Markov network provides both a compact as well as an interpretable representation. The edges in the graph describe which components interact with each other, while the parameters encode the nature of these interactions. This graphical representation has a dual role: edges describe both local influences on the dynamics of neighboring components as well as dependency structure of the equilibrium distribution. In contrast, the graphical structure of CTBN's does not imply about the structure of equilibrium distribution.

In parallel to our work, a compact representation of a the stationary distribution was presented by Yu and Thorne [2006] for the special case of RNA molecules. In this work the stationary distribution is a function of the folding energy of an RNA molecule, which in turn is described as a sum of local terms. Although such representation is a special case of a Markov network, the definition of the dynamics given the folding energy is different than CTMN. The learning procedure presented in this work allows estimation of only global parameters such as a rate scaling factor or the extent to which the equilibrium distribution is affected by the energy function. The language of CTMNs provides a more flexible learning procedure allowing to learn parameters that are associated with local terms. The crux of CTMNs which allows this flexibility is the decomposition of this process into a hidden proposal process and a selection process that depends on a fitness function. The resulting likelihood function is a linear function of hidden sufficient statistics allowing to learn the model using an expectation-maximization procedure.

Exploration of the power of CTMNs to model real life phenomena requires some further steps. As an example we consider the problem of using evolutionary models to learn about structure and function of proteins. Proteins are sequences of basic molecular building blocks called amino-acids of which there are 20 types. The sequence of a protein determines the structure to which it folds, which in turn determines its function. Many species share a common set of proteins such as hemoglobin, which have a relatively conserved structure and function but differ in their sequences [Felsenstein, 2004]. The conserved features of such protein families impose constraints on their sequences. For example, an interaction between two amino acids may occur if they have complementary charges. A major goal in biology is to infer constraints that determine the structure and function of the protein from sequence data.

Learning such constraints from a given set of sequences involves questions such as mapping which positions interact which each other and what is the nature of these interactions. Such constraints can be elegantly modeled as a set of features of Markov network where the parameters account for their quantitative effect on fitness. This approach was recently used to detect residues involved in protein-protein interactions [Weigt et al., 2009, Burger and van Nimwegen, 2010]. However, it uses only correlations within sequences but does not take into account the dynamics of the evolutionary process that generates these sequences. Ignoring this history can lead to severe errors, demonstrating that evolutionary relations is a major confounding factor when determining dependencies between different residues [Bhattacharya et al., 2007]. Burger and van Nimwegen [2010] addressed this problem using a heuristic approach although they argue that the best way to address this difficulty would be to explicitly model the evolution of the sequences along the tree, using an evolutionary model that takes dependencies between positions into account. They claim however that it appears that such a rigorous approach is computationally intractable.

The introduction of CTMN along with the inference methods presented in this thesis are first steps towards tackling this challenge. However there are some additional steps not addressed in this thesis. First, in chapter 2 we presented a parameter estimation procedure. However, learning interactions involves searching for the structure of the stationary distribution of the model that reflect the interactions in the protein. This task is difficult even for Markov networks in the context of static models. However, it should be possible to adopt some of methods used in this domain [Della Pietra et al., 1997, Lee et al., 2007, Ganapathi et al., 2008, Dudı́k et al., 2007]. Second, even if we determine the correct graphical structure of the model, using an exhaustive parameterization for every edge of the graph may be redundant and uninformative. For example, instead of considering every one of the 400 possible amino-acid pairs in a single interaction, considering generic feature such as whether they have complementary charges may be more robust. The CTMN language, in contrast to CTBNs, allows to incorporate such features in a natural manner. However, determining what are the informative features that emerge from data is a challenging task. This problem is related to the structure search problem and has been addressed as well in the context of static Markov networks using semi-heuristic search approaches.

6.2 Gibbs Sampling

The continuous-time Gibbs sampling algorithm proposed here can be described as a block-Gibbs procedure that iterates by sampling trajectories from the distribution over subsets $X_i^{([0,T])}$ given trajectories of other components. Each of theses samples is taken using a dynamic programming procedure, whose complexity scales with the number of transitions.

This algorithm exploits the structure of the model both in CTBNs and in CTMNs. In CTBNs, the posterior distribution of a trajectory of every component depends only on trajectories of its *Markov-blanket*—its parents, children and children's parents. In

CTMNs it depends on trajectories of the neighbors.

An important merit of Gibbs sampling procedures is that they are asymptotically unbiased. Moreover, as our numerical evaluation suggests, in contrast to importance sampling algorithm, the likelihood of the evidence has a relatively mild effect on the convergence rate of the algorithm. These properties make the algorithm suitable for evaluating the bias of more efficient approximation algorithms presented in chapters 4-5.

Two limitations of the algorithm are that in highly coupled networks convergence may be too slow and it is hard to estimate the stopping criteria. To overcome the second problem we experimented with several generic stopping evaluation statistics for such methods [Brooks and Gelman, 1998, Kass et al., 1998] mainly the one of Gelman and Rubin [1992]. However, further experiments are needed to determine how to tune these diagnostic statistics (results not shown).

6.3 Variational Approximations

The mean field and belief propagation algorithms presented in Chapters 4 and 5 are derived from the same variational principle for continuous-time Markov processes. A similar principle and a mean field algorithm was proposed by Opper and Sanguinetti [2007] for a Markov model with indirect and noisy observations. In this work, the posterior distribution of a Markov process is parameterized by a set of continuous functions representing transition rates as a function of time. In Chapter 4 we introduced a different representation—marginal density sets—and reformulate the variational principle.

The marginal density set representation is attractive for two reasons: First, the density set of the exact solution is guaranteed to be bounded. In contrast, in the case of direct evidence, posterior rates tend to infinity near evidence points. Second, this representation is natural in the sense that it is composed of the time derivatives of the expected sufficient statistics that we wish to query. Hence, once we find an optimal solution, these expectations are read out by simple numerical integration. This is analogous to the modern approach for inference in discrete graphical models where instead of representing posterior distributions directly they are represented in terms of expected sufficient statistics [Wainwright and Jordan, 2008].

The differences between the proposed mean field and belief propagation algorithms follows from the structure of the approximation. In the mean field approximation the posterior is constrained to a product of independent inhomogeneous process. This approximation also provides a lower bound to the log-likelihood of evidence allowing to learn parameters of a model using a *variational EM* approach, which searches for parameters that maximize a lower bound of the log-likelihood [Jordan et al., 1999]. Numerical tests suggest that the algorithm provides good results if evidence is not sparse, as is the case in the example of evolutionary models where we are given sequences in the leafs of a phylogenetic tree.

The belief propagation algorithm allows dependencies between different components. The resulting approximation provides highly accurate expected sufficient statistics even in the presence of sparse evidence. The approximation to the log-likelihood is also more accurate than the one of mean field. However, in this case it is not guaranteed to be a lower bound.

Both algorithms search for an optimal approximation using an iterative message passing scheme. In the mean-field algorithm each components use the marginal density sets of neighboring nodes to update its own density set. In belief propagation messages are passed and updated between clusters of nodes. These update steps involve numerical solution of ordinary differential equations, allowing to incorporate standard solvers that use adaptive step size, which in turn provide an optimal trade-off of accuracy versus efficiency.

Prior to our work, Nodelman et al. [2005b] introduced a message passing algorithm for CTBNs that allows dependencies between components. In this work, posterior distribution over clusters of components are modeled using a piecewise homogeneous Markov processes. Saria et al. [2007] demonstrated that the choice of demarcation points between segments of constant parameterization is crucial and suggested a mechanism to automatically tune these points. In contrast, the belief propagation algorithm proposed here uses a non-homogeneous representation equivalent to an interval length that approaches zero. Surprisingly, the resulting algorithm becomes more simple avoiding the need to construct an additional mechanism for adaptive computations, rather than making the problem more complex.

The important insight that belief propagation and mean field algorithm can be derived using similar principles was made by [Yedidia et al., 2005] in the context of discrete models. This insight has led to derivation of a *generalized belief propagation* algorithm and later on to a tremendous number of other extensions [Wainwright et al., 2003, Meshi et al., 2009, Meltzer et al., 2009, Hazan and Shashua, 2010].

Although the belief propagation algorithm provides highly accurate results for tree and toroid topologies, the bias introduced on bidirectional rings motivated us to generalize the continuous-time algorithm. The ring structure induces clusters of triplet components such as $\{X_1, X_2, X_3\}$, $\{X_2, X_3, X_4\}$. While the intersection between such sets include pairs of components (here $\{X_2, X_3\}$) the algorithm requires that the corresponding should agree on each component separately. Generalizing our algorithm to handle stricter constraints, which demand agreement on subsets of components rather than singletons, virtually eliminated this bias (results not shown).

These developments demonstrate that the introduction of marginal density sets, using them to derive a variational principle and using additional tools presented in Chapter 5 can possibly allow to adopt further extensions of belief propagation that are based on similar principles. An additional extension is to derive a more efficient algorithm for CTMNs. Our current approach is to convert the CTMN into a CTBN and to perform inference on the CTBN. This can be inefficient for nodes that have large number of neighbors especially if the cardinality of each component is not small. Performing inference directly on a CTMN can lead to a dramatic improvement in runtime.

Finally, it will be interesting to explore whether the ideas used to develop the continuous-time belief propagation algorithm can be applied to other continuous time models with different state-space structures. Such developments could provide accurate and efficient inference for a wealth of applications in molecular-biology, robotics, social networks, medical care and many more.

6.4 Concluding Remarks

Continuous-time modeling is a promising direction for studying complex dynamic system, although this field of research is its in early stages. In this dissertation I described works aimed to provide tools that facilitate fulfillment of the potential of this field. We showed that by defining an appropriate modeling language, one can potentially capture the main forces of a dynamic process in a compact and interpretable manner. The rest of the thesis deals with the computationally intensive inference task by borrowing approaches from finite dimensional graphical models while exploiting the advantages of continuous time modeling. This requires proper presentations and suitable mathematical tools to exploit them.

A common theme in this thesis is that while the mathematical foundations are somewhat involved, the resulting models and algorithms are relatively simple. The inference algorithms proposed here have a simple flow, which involves standard numeric procedures for manipulation of continuous functions. Additionally, we believe that principled consideration of the dynamic and interactions of a system should lead to more succinct and accurate models avoiding spurious correlations. However, as most of our tests were performed on synthetic data, this conjecture is yet to be supported by experiments with real data. We hope that the accuracy and simplicity that can be obtained by continuous-time modeling will encourage the application of this approach to the wealth of domains that are suitable for this framework.

Bibliography

C. Archambeau, M. Opper, Y. Shen, D. Cornford, and J. Shawe-Taylor. Variational inference for diffusion processes. In *Advances in Neural Information Processing Systems 20*. MIT Press, 2007.

T. Bhattacharya, M. Daniels, D. Heckerman, B. Foley, N. Frahm, C. Kadie, J. Carlson, K. Yusim, B. McMahon, B. Gaschen, S. Mallal, J. I. Mullins, D. C. Nickle, J. Herbeck, C. Rousseau, G. H. Learn, T. Miura, C. Brander, B. Walker, and B. Korber. Founder Effects in the Assessment of HIV Polymorphisms and HLA Allele Associations. *Science*, 315(5818):1583–1586, 2007.

S. P. Brooks and A. Gelman. General methods for monitoring convergence of iterative simulations. *journal of computational graphical statistics*, 7(4):434–455, Dec. 1998.

L. Burger and E. van Nimwegen. Disentangling direct from indirect co-evolution of residues in protein alignments. *PLoS Comput Biol*, 6(1):e1000633, 01 2010.

K. Chung. *Markov Chains with Stationary Transition Probabilities*. Springer Verlag, Berlin, 1960.

I. Cohn, T. El-Hay, N. Friedman, and R. Kupferman. Mean field variational approximation for continuous-time bayesian networks. *Journal of Machine Learning Research*, 11(Oct):2745–2783, 2010.

S. Della Pietra, V. Della Pietra, and J. Lafferty. Inducing features of random fields. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(4):380–393, 1997.

M. Dewar, V. Kadirkamanathan, M. Opper, and G. Sanguinetti. Parameter estimation and inference for stochastic reaction-diffusion systems: application to morphogenesis in d. melanogaster. *BMC Systems Biology*, 4(1):21, 2010.

M. Dudík, S. J. Phillips, and R. E. Schapire. Maximum entropy density estimation with generalized regularization and an application to species distribution modeling. *J. Mach. Learn. Res.*, 8:1217–1260, December 2007.

T. El-Hay, N. Friedman, D. Koller, and R. Kupferman. Continuous time markov networks. In *Proceedings of the Twenty-second Conference on Uncertainty in AI (UAI)*, 2006.

T. El-Hay, N. Friedman, and R. Kupferman. Gibbs sampling in factorized continuous-time markov processes. In *Proceedings of the Twenty-Fourth Conference on Uncertainty in AI (UAI)*, 2008.

T. El-Hay, I. Cohn, N. Friedman, and R. Kupferman. Continuous-time belief propagation. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, 2010.

Y. Fan and C. Shelton. Sampling for approximate inference in continuous time Bayesian networks. In *Tenth International Symposium on Artificial Intelligence and Mathematics*, 2008.

Y. Fan and C. R. Shelton. Learning continuous-time social network dynamics. In *Proceedings of the Twenty-Fifth International Conference on Uncertainty in Artificial Intelligence*, 2009.

Y. Fan, J. Xu, and C. R. Shelton. Importance sampling for continuous time Bayesian networks. *Journal of Machine Learning Research*, 11(Aug):2115–2140, 2010.

J. Felsenstein. *Inferring Phylogenies*. Sinauer, 2004.

V. Ganapathi, D. Vickrey, J. Duchi, and D. Koller. Constrained approximate maximum entropy learning. In *Proceedings of the Twenty-fourth Conference on Uncertainty in AI (UAI)*, 2008.

C. Gardiner. *Handbook of Stochastic Methods*. Springer-Verlag, New-York, third edition, 2004.

A. Gelman and D. B. Rubin. Inference from iterative simulation using multiple sequences. *Statistical Science*, 7(4):457–511, 1992.

A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis, Second Edition (Chapman & Hall/CRC Texts in Statistical Science)*. Chapman and Hall/CRC, 2 edition, July 2003.

W. R. Gilks, R. S., and S. D. J. *Markov Chain Monte Carlo in Practice*. Chapman & Hall, 1996.

K. Gopalratnam, H. Kautz, and D. S. Weld. Extending continuous time bayesian networks. In *AAAI'05: Proceedings of the 20th National Conference on Artificial Intelligence*, pages 981–986. AAAI Press, 2005.

T. Hazan and A. Shashua. Norm-product belief propagation: Primal-dual message-passing for approximate inference. *Information Theory, IEEE Transactions on*, 56 (12):6294 –6316, Dec. 2010.

M. I. Jordan, Z. Ghahramani, T. Jaakkola, and L. K. Saul. An introduction to variational approximations methods for graphical models. In M. I. Jordan, editor, *Learning in Graphical Models*. MIT Press, Cambridge MA, 1999.

R. E. Kass, B. P. Carlin, G. A., and R. M. Neal. Markov chain monte carlo in practice: A roundtable discussion. *The American Statistician*, 52:93–100, 1998.

D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, Cambridge, MA, 2009.

S.-I. Lee, V. Ganapathi, and D. Koller. Efficient structure learning of Markov networks using L1-regularization. In *Advances in Neural Information Processing Systems (NIPS 2006)*, 2007.

T. Meltzer, A. Globerson, and Y. Weiss. Convergent message passing algorithms - a unifying view. In *Proc. Twenty-fifth Conference on Uncertainty in Artificial Intelligence (UAI)*, 2009.

O. Meshi, A. Jaimovich, A. Globerson, and N. Friedman. Convexifying the bethe free energy. In *Proc. Twenty-fifth Conference on Uncertainty in Artificial Intelligence (UAI)*, 2009.

N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of State Calculations by Fast Computing Machines. *J. Chem. Phys.*, 21:1087–1092, June 1953.

K. P. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, University of California, Berkeley, 2002.

B. Ng, A. Pfeffer, and R. Dearden. Continuous time particle filtering. In *Proc. of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI)*, 2005.

U. Nodelman, C. Shelton, and D. Koller. Continuous time Bayesian networks. In *Proc. Eighteenth Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 378–387, 2002.

U. Nodelman, C. Shelton, and D. Koller. Learning continuous time Bayesian networks. In *Proc. Nineteenth Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 451–458, 2003.

U. Nodelman, C. Shelton, and D. Koller. Expectation maximization and complex duration distributions for continuous time Bayesian networks. In *Proc. Twenty-first Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 421–430, 2005a.

U. Nodelman, C. Shelton, and D. Koller. Expectation propagation for continuous time Bayesian networks. In *Proc. Twenty-first Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 431–440, 2005b.

M. Opper and G. Sanguinetti. Variational inference for Markov jump processes. In *Advances in Neural Information Processing Systems 20*. MIT Press, 2007.

M. Opper and G. Sanguinetti. Learning combinatorial transcriptional dynamics from gene expression data. *Bioinformatics*, 26(13):1623–1629, 2010.

S. Rajaram, T. Graepel, and R. Herbrich. Poisson-networks: A model for structured point processes. In *Proc. Tenth International Workshop on Artificial Intelligence and Statistics (AISTATS)*, January 2005.

A. Ruttor and M. Opper. Approximate parameter inference in a stochastic reaction-diffusion model. In *Proc. Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 9, pages 669–676, 2010.

G. Sanguinetti, A. Ruttor, M. Opper, and C. Archambeau. Switching regulatory models of cellular stress response. *Bioinformatics*, 25(10):1280–1286, 2009.

S. Saria, U. Nodelman, and D. Koller. Reasoning at the right time granularity. In *Proc. Twenty-third Conference on Uncertainty in Artificial Intelligence (UAI)*, 2007.

A. Simma, M. Goldszmidt, J. MacCormick, P. Barham, R. Black, R. Isaacs, and R. Mortier. Ct-nor: Representing and reasoning about events in continuous time. In *Proc. Twenty-fourth Conference on Uncertainty in Artificial Intelligence (UAI)*, 2008.

M. J. Wainwright and M. Jordan. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.*, 1:1–305, 2008.

M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. Tree-based reparameterization framework for analysis of sum-product and related algorithms. *IEEE Transactions on Information Theory*, 49:2003, 2003.

C. Wang, D. Blei, and D. Heckerman. Continuous Time Dynamic Topic Models. In *Proceedings of the 25th International Conference on Machine Learning (ICML)*, 2008.

M. Weigt, R. A. White, H. Szurmant, J. A. Hoch, and T. Hwa. Identification of direct residue contacts in protein-protein interaction by message passing. *Proceedings of the National Academy of Sciences of the United States of America*, 106(1):67–72, January 2009.

J. S. Yedidia, W. T. Freeman, and Y. Weiss. Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Transactions on Information Theory*, 51(7):2282–2312, 2005.

J. Yu and J. L. Thorne. Dependence among sites in RNA evolution. *Mol. Biol. Evol.*, 23:1525–37, 2006.

שאילתות. נראה כיצד לבצע צעדי דגימה אלו באופן יעיל שבו זמן החישוב תלוי בקצב השנויים בתהליך. בעוד שקשה לחזות את זמן הריצה מראש, לכייל את תנאי העצירה או לאמוד את שגיאת הקירוב של האלגוריתם, זהו האלגוריתם הראשון שנותן דגימות בלתי מוטות אסימפטוטית ב-CTBNs.

גישה נוספת לפיתוח קירובים היא שימוש בקירובים וריאציונים אשר בהם ההתפלגות הא-פוסטריורית מיוצגת על ידי התפלגויות פשוטות יותר המאפשרות לענות על שאילתות בצורה יעילה. במטרה לאמץ גישה זו, נראה שניתן לייצג התפלגויות אלה באמצעות אוסף פונקציות רציפות תלויות-זמן. הייצוג זה יאפשר לנו לפתח גישת mean-field חדשה שמקרבת את ההתפלגות הא-פוסטריורית על ידי אוסף של תהליכים מרקובים בלתי תלויים. בעוד שהנחה זו מובילה להטייה מסויימת, האלגוריתם המתקבל הוא יעיל ומהיר. יותר מכך, האלגוריתם מחשב חסם תחתון ללוג-הניראות שהינו בעל חשיבות בלמידה של פרמטרים שבה אנו מנסים למקסם את הניראות. באמצעות גישה זו מתקבל אלגוריתם אלגנטי שבו רכיבים שונים מעבירים בינהם מידע על ההתפלגות המיוצג באמצעות אוסף של פונקציות רציפות. הרכיבים השונים מעבדים מידע זה על ידי פתרון של מערכות של משוואות דיפרנצאיליות רגילות באמצעות שיטות נומריות סטנדרנטיות. מאחר ושיטות אלו לאינטגרצייה נומרית משתמשות בצעדי זמן אדפטיביים, סיבוכיות ההסקה היא נמוכה ברכיבים ובקטעי זמן שבהם הדינמיקה היא יוניפורמית.

הייצוג החדש של התפלגויות א-פוסטריוריות מאפשר לבחון קירובים וריאציונים עשירים יותר מ-mean-field ועל ידי כך להקטין את הטיית הקירוב. באופן זה נפתח אלגוריתם belief-propagation המאפשר לבצע הסקה יעילה תוך התחשבות בתלויות בין רכיבים שונים אל ידי יצוג של התפלגויות משותפות של תתי-תהליכים של קבוצות חופפות של רכיבים. באופן דומה ל-mean-field מתקבל אלגוריתם יעיל שמשתמש באינטגרצייה נומרית של פונקציות רציפות. בדיקות אמפיריות מראות שיפור משמעותי יחסית לאלגוריתם ה-mean-field כך שמתקבלות תוצאות בעלות דיוק גבוה על מגוון של מודלים.

תקציר

מערכות דינימיות רבות, בתחומי מחקר מגוונים, ניתנות לתיאור באופן טבעי כתהיליכים שמתפתחים בזמן-רציף. לעתים קרובות מערכות כאלה מורכבות מרכיבים רבים בעלי אינטראקציות בינהם, כאשר כל רכיב משנה את מצבו בקצב שונה מהאחרים. בשנים האחרונות התפתח אפוא תחום מחקר העוסק בשאלה כיצד ניתן ללמוד ולהבין מערכות דינמיות מרובות רכיבים המתפתחות בזמן-רציף.

אחת הגישות הנפוצות ללימוד מערכות שכאלה היא בניית מודל הסתברותי על סמך תצפיות אמפיריות. גישה זו מצריכה שלושה מרכיבים עיקריים : **שפת מידול** המתארת את התכונות העיקריות של התהליך באופן ממצה, שיטות **למידה** המאפשרות לאמוד פרמטרים ומבנה של מודל ספציפי מתצפיות אמפיריות ושיטות **הסקה** המאפשרות הן לבצע תחזיות בעזרת המודל לצד נתונים זמינים והן לחשב נראות של תצפיות אמפיריות. דוגמה לגישה שכזו במערכות דינמיות היא שפת המידול החדשה הנקראת **רשתות ביסיאניות בזמן-רציף** (continuous-time Baysian networks) או בקיצור CTBNs. שפה זו מתארת באופן קומפקטי תהליך מרקובי שבו לכל רכיב יש אינטראקציות עם מספר קטן יחסית של רכיבים אחרים. מאחר והמחקר על מודלים מהסוג הזה עדיין בחיתוליו, קיים עדיין פער גדול בין שפע שפות המידול, אלגוריתמי הלמידה ושיטות ההסקה בתחומיים מסורתיים לאלו הקיימות בתחום הזמן הרציף.

בעבודה מחקרית זו אנחנו עושים להרחבת אוסף היישומים שניתן לנתח בזמן רציף באמצעות פיתוח שלושה מרכיבים הנדרשים לכך. תחילה, נגדיר שפת מידול חדשה הנקראת **רשתות מרקוביות בזמן-רציף** (continuous-time Markov networks) או בקיצור CTMNs. שפה זו מתאימה במיוחד לחקר תהליכים באבולוציה מולקולרית בהם הדינמיקה היא תוצר של יחסי גומלין בין שני כוחות מנוגדים : מוטציות אקראיות של רכיבים בודדים (נוקלאוטידים במולקולות DNA ו-RNA או חומצות אמיניות בחלבונים) וכוחות ברירה טבעית הפועלים על הרצף כולו. אחר כך, נפתח ונציג שיטה איטרטיבית ללימוד המודל מתצפיות אמפיריות, כאשר בכל איטרציה עלינו לפתור את בעיית ההסקה. בנוסף, נראה שאת בעיית ההסקה ב-CTMNs ניתן לנסח כבעייה שקולה ב-CTBNs.

מאחר ובעיית ההסקה ב-CTBNs מהותית גם בלמידה של CTBNs וגם בלמידה של CTMNs, נתמקד בבעיה זו בשאר חלקי המחקר. נפתח אפוא שלושה אלגורתמים להסקה מקורבת שלכל אחד מהם תכונות שונות המשלימות זו את זו. אלגוריתמים אלו מאמצים תובנות מתחום ההסקה במודלים הסתברותיים ממימד סופי תוך ניצול יתרונות של מידול בזמן רציף, המאפשרים פיתוח שיטות יעילות ופשוטות יחסית כאשר מספר פעולות החישוב שהן מפעילות תלוי בצורה פשוטה בדינימיקה של התהליך.

אלגוריתם ההסקה הראשון אותו נציג מבוסס על אסטרטגייה של דגימת גיבס (Gibbs sampling). האלגוריתם דוגם מסלולים מההתפלגות הא-פוסטרירית בהינתן התצפיות ומשתמש בדגימות אלה כדי לענות על

עבודה זו נעשתה בהדרכתו של

פרופ׳ ניר פרידמן

הסקה הסתברותית במערכות דינמיות מורכבות בזמן-רציף

חיבור לשם קבלת תואר דוקטור לפילוסופיה

מאת

טל אל-חי

הוגש לסנאט האוניברסיטה העברית בירושלים

ינואר 2011